PanaXSeries

The One to Watch for Constant Innovation-Making the Future Come Alive

MICROCOMPUTER MN102H00

MN102H55D/55G/F55G

LSI User's Manual

Pub.No.22355-020E

Panasonic

The MN102HF55G is manufactured and sold under the License Agreement with BULL CP8 Inc., and the use of the MN102HF55G into the IC card is not allowed.

Request for your special attention and precautions in using the technical information and semiconductors described in this manual.

(1) The approval of the Japanese Government is required for the export of any products and technologies listed in this manual which are subjected to the provisions of the Foreign Exchange and Foreign Trade Law.

(2) The contents of this manual are subject to change without notice to improve design, function, or performance.

(3) Matsushita Electronics assumes no responsibility or liability for damages or for infringements of patents or other rights arising from use of the information in this manual.

(4) The contents of this manual may not be copied or reproduced without permission in writing from Matsushita Electronics.

(5) This manual describes standard specifications. Obtain the latest product standard specifications before you design, purchase, or use.

For inquiries regarding this manual or any Matsushita semiconductor, please contact one of the sales offices listed at the end of this manual or the sales department of Matsushita Electronics Corporation.

About This Manual

This manual is intended for assembly-language programming engineers. It describes the internal configuration and hardware functions of the MN102H55D/55G/F55G microcontrollers.

Text Conventions

This manual contains titles, sub-titles, special notes and warnings. Supplementary comments appear in the sidebar.

Warning
Please read and follow these instructions to prevent damage or reduced performance.

Finding Desired Information

This manual provides four methods for finding desired information quickly and easily.

(1) An index for the front of the manual for finding each section.

(2) A table of contents at the front of the manual for finding desired titles.

(3) A list of figures at the front of the manual for finding illustrations and charts by names.

(4) A chapter name is located at the upper corner of each page.

Related Manuals

■ MN10200 Series Linear Addressing High-speed Version LSI User Manual
(Describes the MN10200 series specifications)

■ MN10200 Series Linear Addressing High-speed Version Instruction Manual
(Describes the instruction set)

■ MN10200 Series Linear Addressing High-speed Version C Compiler User Manual Usage Guide (Describes the installation, commands, and options for the C complier)

■ MN10200 Series Linear Addressing High-speed Version C Compiler User Manual Language Description (Describes the syntax for the C complier)

■ MN10200 Series Linear Addressing High-speed Version C Compiler User Manual Library Reference (Describes the standard libraries for the C complier)

■ MN10200 Series Linear Addressing High-speed Version Cross Assembler User Manual Language Description (Describes the assembler syntax and notation)

■ MN10200 Series Linear Addressing Version C Source Code Debugger User Manual
(Describes the use of the C source code debugger)

■ MN10200 Series Linear Addressing Version PanaXSeries Installation Manual
(Describes the installation of the C complier, cross-assembler, and C source code debugger and the procedures for using the in-circuit emulator)

Questions and Comments

Please send your questions, comments and suggestions to the semiconductor design center closest to you. See the last page of this manual for a list of addresses and telephone numbers.

Contents

Chapter 1 General Description

Chapter 2 Bus Interface

Chapter 3 Interrupts

Chapter 4 Timers

Chapter 5 Serial Interface

Chapter 6 Analog Interface

Chapter 7 ATC, ETC

Chapter 8 Ports

Chapter 9 System Control

Chapter 10 Low-power Modes

Chapter 11 Appendix

List of Figures

List of Tables

Chapter 1 General Description

1

1-1 General Description

1-1-1 Introduction

The 16-bit MN102 series high-speed linear addressing version designs the new architecture for C-language programming based on a detailed analysis for embedded applications. This improves the previous system architecture in speed and function to meet the requirements in user systems including miniaturization to power consumption.

This series adapts a load/store architecture method for computing within registers instead of the accumulator system for computing within the memory space in the previous series. The basic instructions are one byte/one machine cycle. This reduces code size and improves compiler efficiency. This series uses the circuit designed for submicron technology providing optimized hardware and low system power consumption.

This series has up to 16 Mbytes of linear address space and can develop the highly efficient programs. The optimized hardware architecture allows lower power consumption even in large systems.

1-1-2 Features

This series contains a flexible and optimized hardware architecture as well as a simple and efficient instruction set. This allows economy and speed. This section describes the features of this series CPU.

1. High-speed Signal Processing

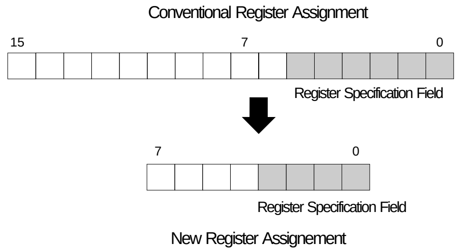
An internal multiplier operates 16-bit x16-bit = 32-bit in a single cycle. In addition, the hardware contains a saturation calculator which must be used in signal processing and increases the signal processing speed.

2. Linear Addressing for Large Systems

The MN102H series contains up to 16 Mbytes of linear address space. The CPU does not detect borders between address spaces, which provides an effective development environment. The hardware architecture is also optimized for large systems. The memory is not divided into instruction areas and data areas so that operations can share instructions.

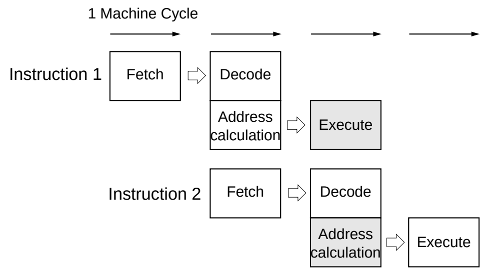
3. Single-byte Basic Instruction Length

The MN102H series has replaced general registers with eight internal CPU registers divided four address registers (A0-A3) and four data registers (D0-D3). The register specification fields are four bits or less, and the code size of the basic instructions including register-to register operations and load/store operations is one byte.



4. High-speed Pipeline Processing

The MN102H series executes instructions in a 3-stage pipeline: fetch, decode, execute. This allows the MN102H series to execute instructions of single byte in one machine cycle.

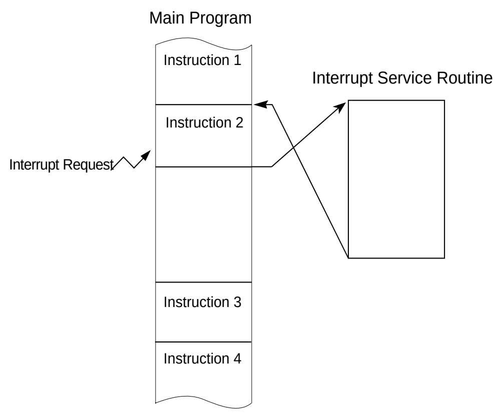


5. Simple Instruction Set

The MN102H series uses an instruction set of 41 instructions, designed specially for the programming model for embedded applications. To shrink code size, instructions have a variable length of one to seven bytes. The most frequently used instructions in C-language compiler are single byte.

6. High-speed Interrupt Response

The MN102H series halts the instructions execution even during the execution of the instruction with long execution cycles. After an interrupt occurs, the program moves to the interrupt service routine within six cycles or less. The MN102H series improves real-time control performance using the interrupt handler which adjusts interrupt servicing speed.



7. Flexible Interrupt Control Structure

The interrupt controller supports a maximum of 64 interrupt vectors (of them, interrupt vectors 0 to 3 are reserved for nonmaskable interrupts). In addition, groups of up to four vectors are assigned to classes. Each class can be set to one of seven priority levels. This provides the software design flexibility and control. The CPU is compatible with software from previous Panasonic peripheral modules.

8. High-speed, High-functional External Interface

The MN102H series supports external interface functions including DMA, handshake function and bus arbitration.

9. C-language Development Environment

The MN102H series has simple hardware optimized for C-language programming and highly efficient C compiler. With this advantage, this series improves development environment for C-language embedded applications without expanding the program size. The **PanaXSeries** development tools support the MN102H series devices.

10. Outstanding Power Savings

The MN102H series contains separate buses for instructions, data and peripheral functions, which distribute and reduce load capacitance. This reduces overall power consumption. The MN102H series also supports three modes of SLOW, HALT and STOP for power savings.

1-1-3 Overview

This section describes the basic configuration and functions of the MN102H55D/55G/F55G.

Note 1: These bits change depending on all 24 bits of the operation result.
Note 2: These bits change depending on the lower 16 bits of the operation result.

The IE flag should be set to 0 before the IM[2:0] flags of PSW are changed.

■ Processor Status Word (PSW)

The PSW register contains the operating result flags and interrupt mask level flags.

	15							8	7 (Note 1)				(Note 2)			0
PSW	-	ST	S1	S0	IE	IM2	IM1	IM0	VX	CX	NX	ZX	VF	CF	NF	ZF
Reset :	-	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Zero flag (ZF) is set if the lower 16 bits of the operation result are 0; otherwise it is reset.

Negative flag (NF) is set if bit 15 of the operation result is 0; otherwise it is reset.

Carry flag (CF) is set if the operation resulted in a carry or a borrow out of bit 15; otherwise it is reset.

Overflow flag (VF) is set if the operation causes the sign bit to change in a 16-bit signed number; otherwise it is reset.

Extension zero flag (ZX) is set if all bits of the operation result are 0; otherwise it is reset.

Extension negative flag (NX) is set if the MSB of the operation result is 1, and it is reset if the MSB is 0.

Extension carry flag (CX) is set if the operation resulted in a carry or a borrow out of the MSB; otherwise it is reset.

Extension overflow flag (VX) is set if the operation causes the sign bit to change in a 24-bit signed number; otherwise it is reset.

IM[2:0] indicate the mask level (from 0 to 7) of interrupts that the CPU will accept from its seven interrupt input pins. The CPU will not accept any interrupt at a higher level than the indicated level here.

Interrupt enable flag (IE) controls maskable interrupt enable. The flag is set if IE=1, and it is reset if IE=0.

S[1:0] are OS software control bits. These are reserved for the OS.

Saturation flag controls whether or not the CPU performs a saturation operation. When this bit is 1, the CPU execute a saturate operation. When this bit is 0, the CPU operates a normal operation. The PXST instruction can reserve the meaning of this bit for the next instruction.

Figure 1-1-1 Processor Status Word (PSW)

Please refer to "11-5 MN102H00 series High-speed Linear Address Instruction Set" for the flags reflected by instructions.

■ Internal Registers, Memory, and Special Function Registers

Program Counter

23	0
PC	

The program counter specifies the 24-bit address of the program during the execution.

Address Registers

23	0
A0	
A1	
A2	
A3	

The address registers specify the data location on memory. Of four registers, A3 is assigned as the stack pointer.

Data Registers

23	0
D0	
D1	
D2	
D3	

The data registers perform all arithmetic and logic operations. When the byte-length (8-bit) or word-length (16-bit) data is transferred to memory or to another register, an instruction adds a zero or sign extension.

Multiplication/Division Register

15	0
MDR	

The multiplication/division register stores the upper 16 bits of the 32-bit product of multiplication operations. In division operations, this register stores the upper 16 bits of the 32-bit dividend before execution, and the 16-bit remainder of the quotient after execution.

Processor Status Word

15	0
PSW	

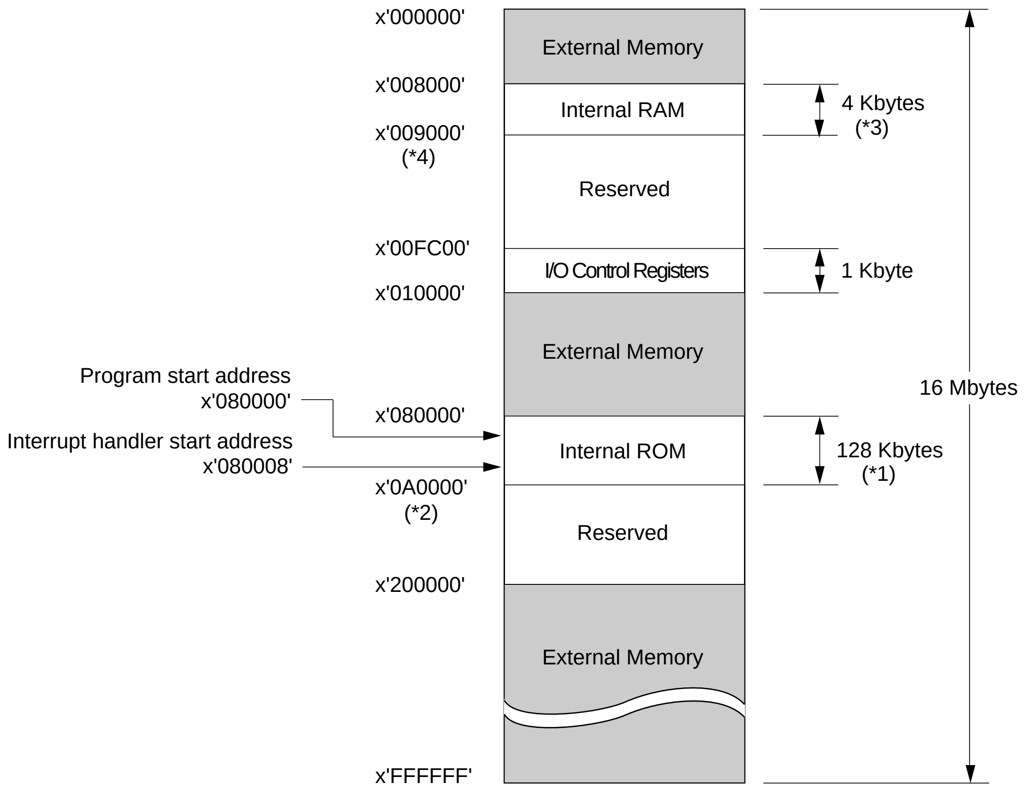
Memory, Special Function Registers, I/O Ports

ROM	Memory (ROM, RAM), special function registers for peripheral function control and I/O ports are assigned to the same address space.
RAM	
CPUM, EFCR, IAGR	Internal Control Registers *
NMICR, xxICR	Interrupt Control Registers *
SCCTRn, TRXBUFn, SCSTRn	Serial Interfacel Registers *
ANCTR, ANnBUF	A/D Converter Registers *
TMn, BCn, BRn	Timer/Counter Registers *
MEMMD	Memory Control Registers *
PnOUT, PnIN, PnDIR	I/O Port Registers *

* This allocation is an example. Actual memory, peripheral functions, special function registers and I/O port allocation depends on the model.

■ Address Space

The memory contains up to 16 Mbytes of linear address space. The instruction and data areas are not separated, so that the internal RAM, special function registers for internal peripheral functions are allocated into the first 64 kbytes in memory as the basic configuration. There are three memory modes as following depending on models.



Note*) Parameters on the right table differ on each chip model.

	*1	*2	*3	*4
MN102HF55G	128 KBytes	x'0A0000'	4096 Bytes	x'009000'
MN102H55G	128 KBytes	x'0A0000'	4096 Bytes	x'009000'
MN102H55D	64 KBytes	x'090000'	4096 Bytes	x'009000'

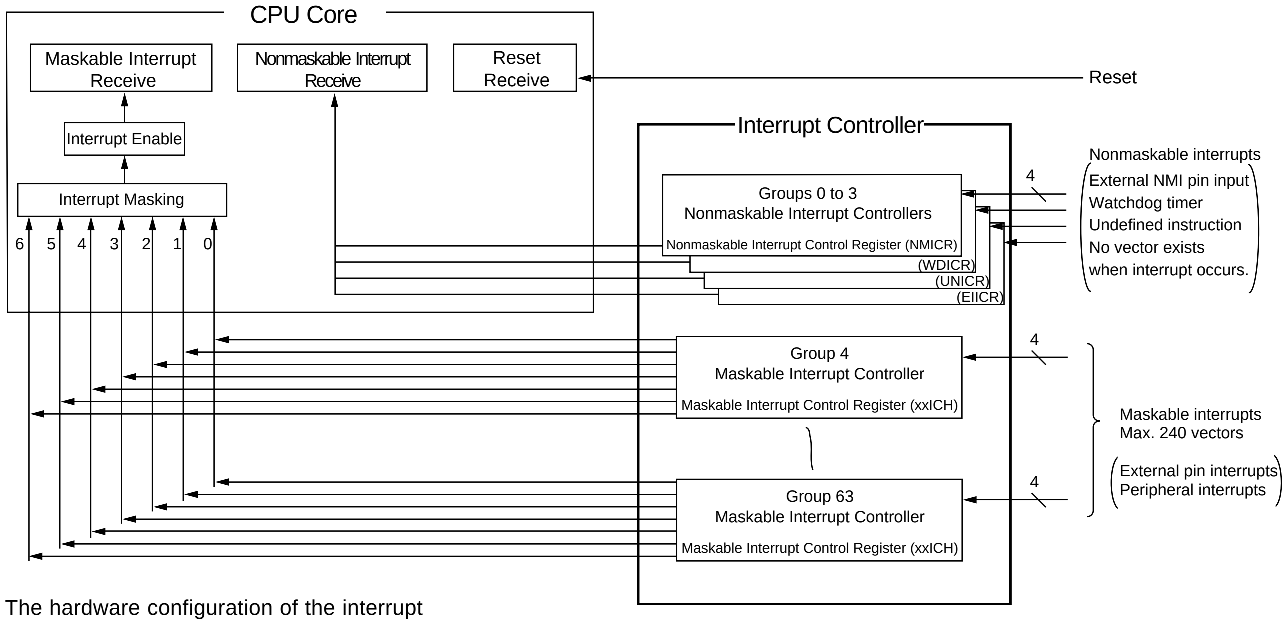
Figure 1-1-2 Address Space (Memory Expansion Mode)

Table 1-1-1 Memory Modes

Mode	Address Bit Width	ROM Capacity	External Memory Access
Single-chip mode	————	64 k/128 kbytes	Not accessible
Memory expansion mode	Up to 24 bits		Accessible
Processor mode		None	

■ Interrupt Controller

The interrupt controller allocated to the outside of the core controls all nonmaskable and maskable interrupts except reset. Each class has up to four interrupt vectors and specifies any of seven priority levels.



The hardware configuration of the interrupt controller depends on the model.

Figure 1-1-3 Interrupt Controller Configuration

The CPU checks the processor status word to determine whether an interrupt request is accepted or not. If an interrupt is accepted, automatic servicing by hardware starts and the program counter and PSW are pushed to the stack. Next, the program moves to interrupt, searches the interrupt vector and branches to the entry address of the interrupt service routine for that interrupt.

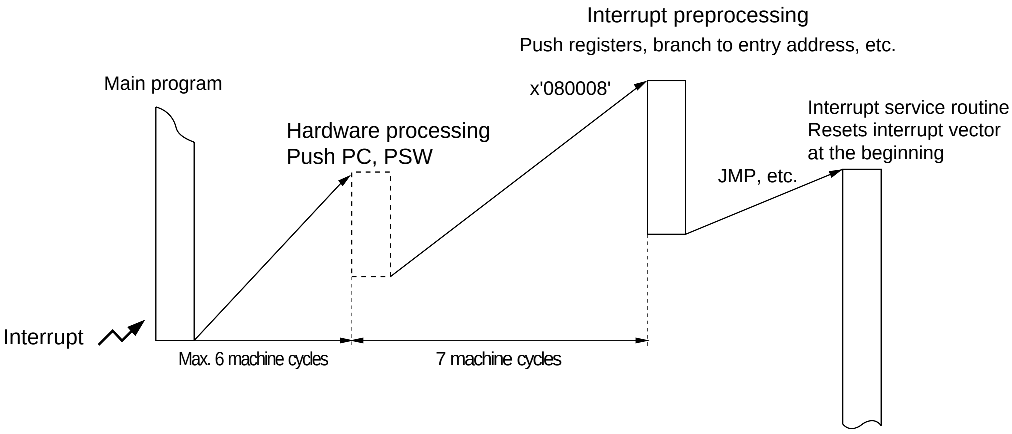


Figure 1-1-4 Interrupt Servicing Sequence

1-2 Basic Specification

Table 1-2-1 Basic Specifications

Structure	Internal multiplier (16-bit×16-bit=32-bit) and internal saturate operation calculator
	Load/store architecture
	Eight registers: Four 24-bit data registers Four 24-bit address registers
	Others: 24-bit program counter 16-bit processor status word 16-bit multiplication/division register
Instruction Set	41 instructions
	6 addressing modes
	1-byte basic instruction length
	Code assignment: 1 byte (basic) + 0 to 6 bytes (extension)
Performance	Maximum of 17-MHz internal operating frequency with a 34-MHz external oscillator MN102H55D/55G Maximum of 15-MHz internal operating frequency with a 30-MHz external oscillator MN102HF55G
	Instruction execution clock cycles: For register-to-register operations, minimum 1 cycle For branch operations, minimum 2 cycles For load/store operations, minimum 1 cycle
Pipeline	3 stage: instruction fetch, decode, execute
Address Space	Up to 16-Mbyte linear address space Shared instruction/data space
Interrupt	1 external nonmaskable interrupt 46 maskable interrupts 7 priority level settings
Low-power Mode	SLOW, STOP, HALT
Oscillation Frequency	Up to 34 MHz
Timer/Counter	Eight 8-bit timers (TM0 to TM7) : Cascading function (form as 16-bit to 64-bit timer) Timer output Internal clock source or external clock source Serial Interface clock generation Start timing generation for A/D converter

Table 1-2-1 Basic Specifications

Timer/Counter	Five 16-bit timers (TM8 to TM12) : Two channels of compare/capture registers Selectable internal or external clock PWM/one-shot pulse output Two-phase encoder input (4x or 1x method)
	Two 8-bit PWM (TM13, TM14) : Two internal compare registers for each channel Two pattern outputs
	One 16-bit pulse width counter (TM15) : Capture the counter value whenever the input pulse rises
	16-bit watchdog timer
ATC	Four Channels Automatic transfer is possible between memories, memory and peripheral I/O for each interrupt vector. Transfer unit: byte or word Transfer mode: single-chip or burst mode Transfer addressing: source, destination pointer, increment Up to 4096 words can be transferred Access to 16-Mbyte address space
ETC	Two Channels Automatic transfer is possible between external device and external memory. Transfer unit: byte or word Transfer mode: single-chip or burst mode Transfer addressing: source, destination pointer, increment Up to 4096 words can be transferred Access to 16-Mbyte address space
Serial Interface	Three Synchronous Interfaces (ASCI0 to ASCI2) Two shared UART/Synchronous/I²C (single master only) Interfaces (ASCI3, ASCI4)
A/D Converter	10-bit with 8 channels (can be used as 8-bit) Automatic Scanning
D/A Converter	Two 8-bit channels
External Expansion	Address/data multiplex port function Address/data separate port function
Memory Interface	DRAM Interface (8-bit/16-bit width) Burst ROM Interface
I/O Port	Maximum of 82 I/O ports in single-chip mode Maximum of 47 I/O ports in address/data multiplex mode Maximum of 40 I/O ports in address/data separate mode
Package	100-pin LQFP

1-3 Block Diagram

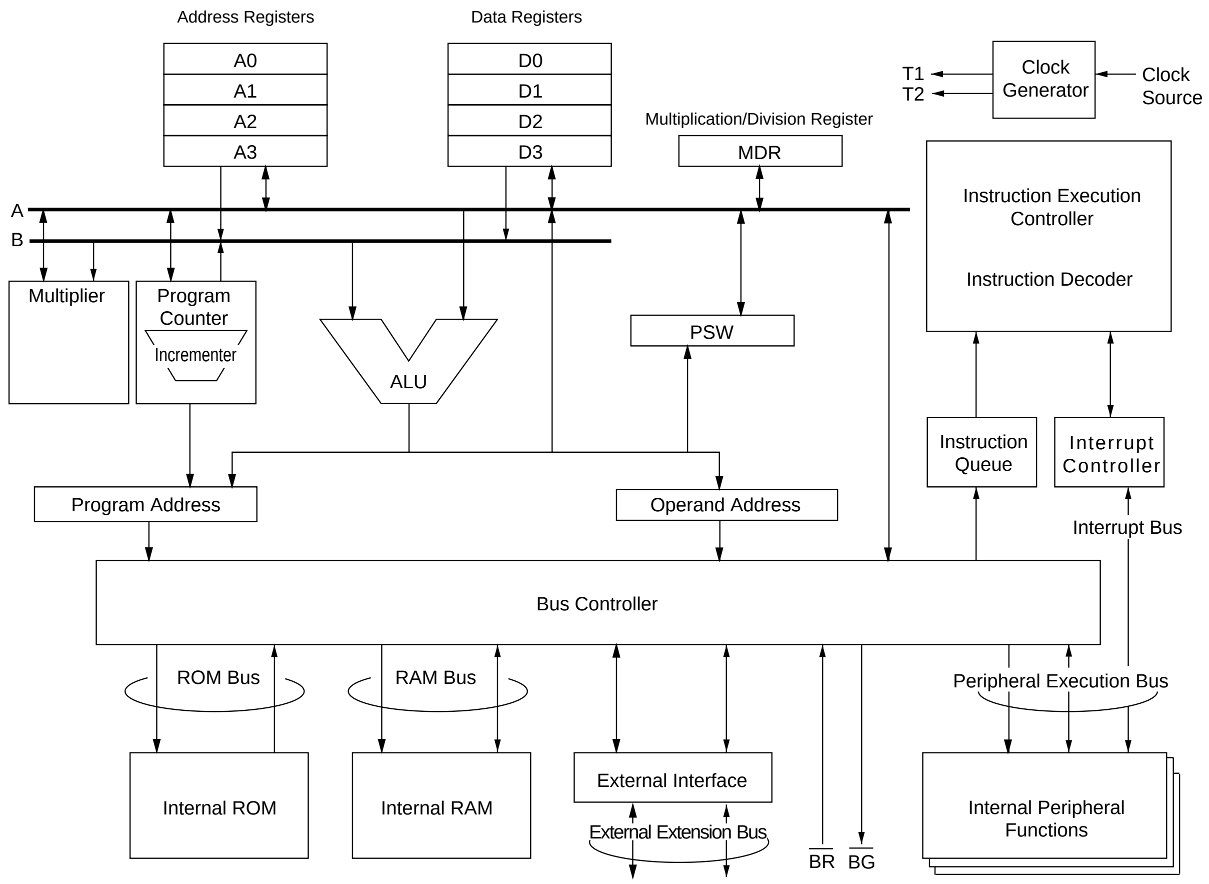


Figure 1-3-1 Block Diagram

Table 1-3-1 Block Functions

Blcok	Function
Clock Generator	An oscillation circuit connected to an external crystal supplies the clock to all blocks in the CPU.
Program Counter	The program counter generates addresses for instruction queues. Normally it increments based on the sequencer indications, but for branch instructions and interrupt acceptance, it sets the branch address and the ALU operation result.
Instruction Queue	The instruction queue contains up to four bytes of prefetched instructions.
Instruction Decoder	The instruction decoder decodes the contents of instruction queue and generates control signals needed for the instruction execution. The instruction executes by controlling each block in the CPU.
Quick Decoder	The quick decoder decodes the 2-byte or larger instruction at faster speed.
Instruction Execution Controller	The instruction execution controller controls the CPU operations based on results from the instruction decoder and interrupt requests.
ALU	The ALU calculates the operand addresses for arithmetic operations, logic operations, shift operations, register relative indirect addressing, indexed addressing, register indirect addressing.
Multiplier	The multiplier calculates 16 bits $\times$ 16 bits = 32 bits.
Internal ROM and RAM	These memory allocate the program, data and stack areas.
Address Registers (An)	The address registers (An) store the addresses in memory accessed during data transfer. They also store the base addresses in register relative indirect addressing, indexed addressing and register indirect addressing modes.
Operation Registers	The data registers (Dn) store data transferred to memory and results of arithmetic operations. They also store the offset addresses in indexed addressing and register indirect addressing modes. The multiplication/division register (MDR) stores data for multiplication/division operations.
PSW	The processor status word register stores the flags that indicate the status of the CPU interrupt controller and operation results.
Interrupt Controller	The interrupt controller detects interrupt requests from the peripheral functions, and requests the CPU to move to the interrupt servicing routine.
Bus Controller	The bus Controller controls the connection between the CPU internal bus and the CPU external bus. It also contains the bus arbitration function.
Internal Peripheral Function	The MN102H55D/55G/F55G contains internal peripheral functions including timers, serial interface, A/D converter and D/A converters. Internal peripheral functions vary depending on the chip models.

1-4 Pin Description

1-4-1 Single-chip Mode

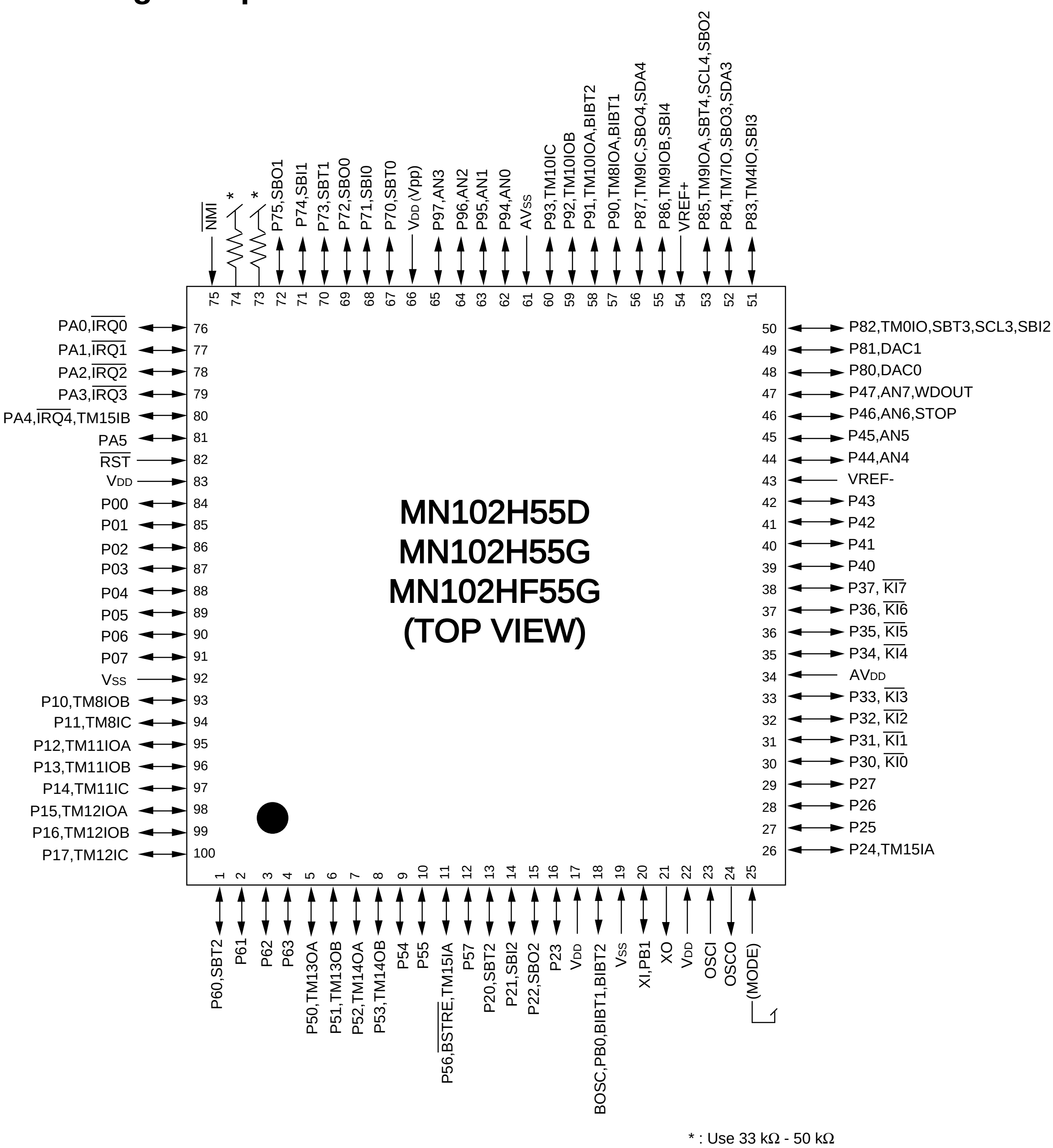


Figure 1-4-1 Pin Configuration in Single-chip Mode

Unused pins require handling in the circuit (input pins are connected to VDD/VSS, output pins leave open, input/output pins are connected to VDD/VSS or leave open depending on pin direction setting).

1-4-2 Memory Expansion Mode with 8-bit Bus Address/ Data Separate Mode

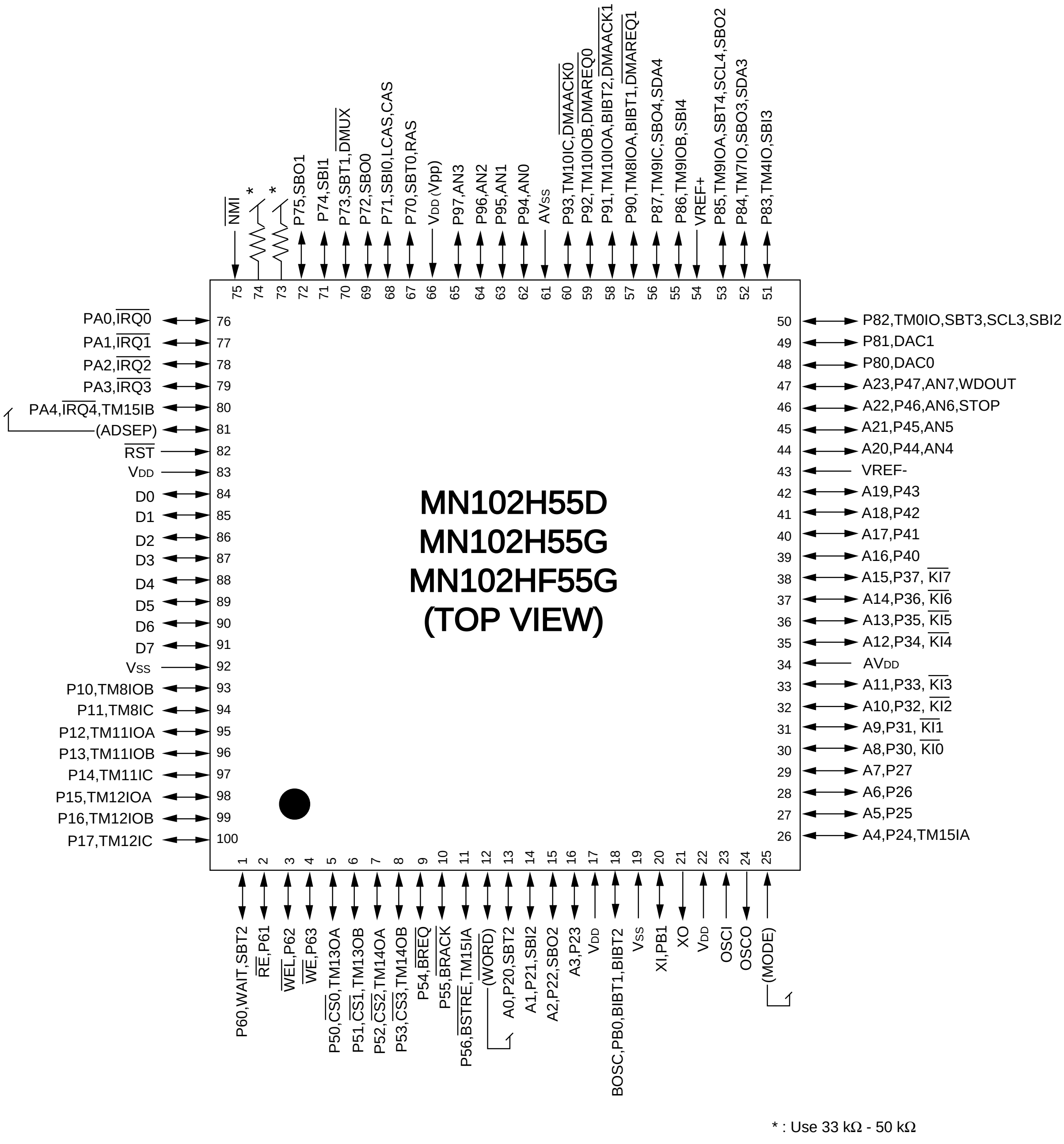


Figure 1-4-2 Pin Configuration in Memory Expansion Mode with 8-bit Bus Address/Data Separate Mode

Unused pins require handling in the circuit (input pins are connected to VDD/VSS, output pins leave open, input/output pins are connected to VDD/VSS or leave open depending on pin direction setting).

1-4-3 Memory Expansion Mode with 16-bit Bus Address/Data Separate Mode

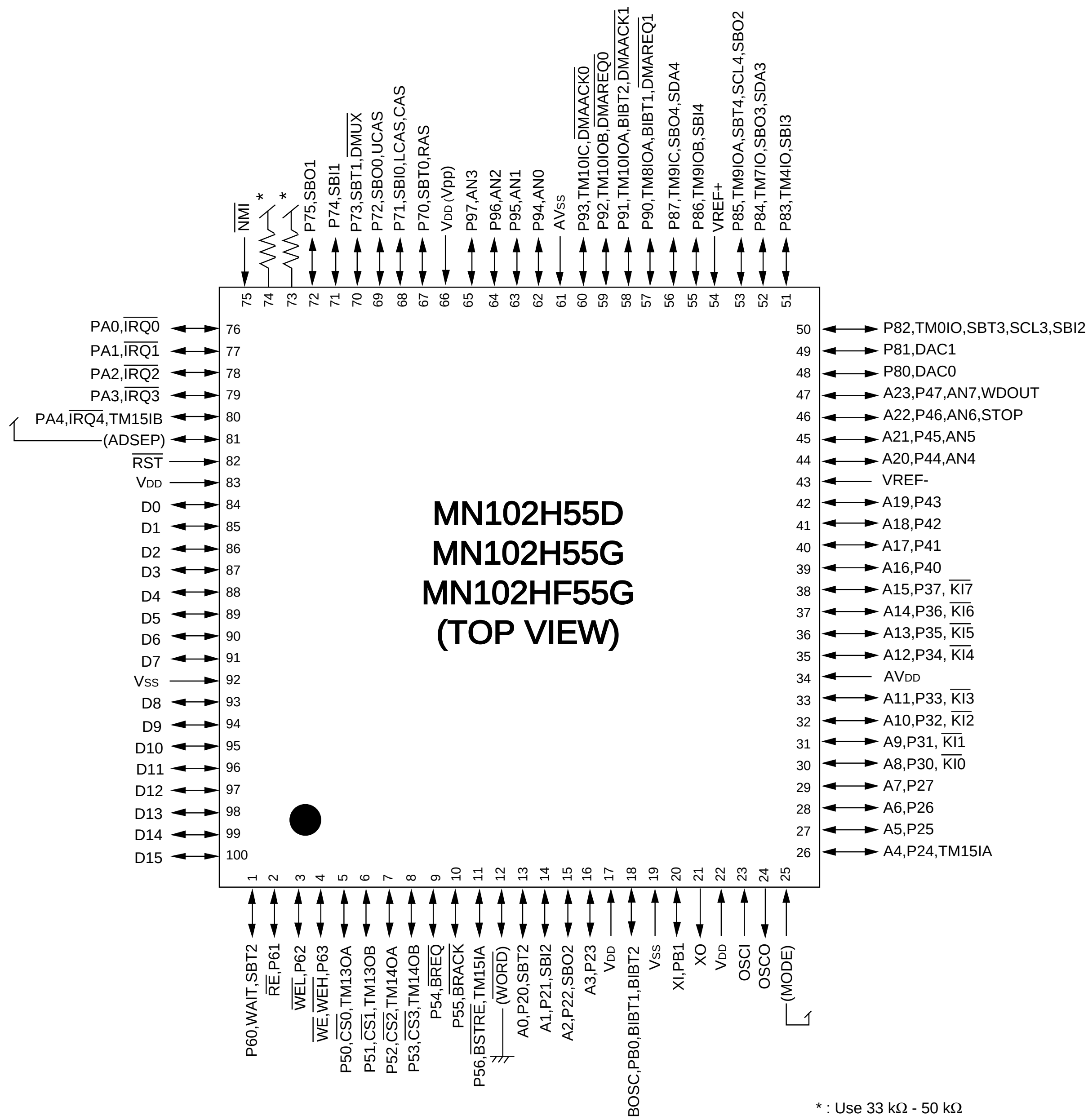


Figure 1-4-3 Pin Configuration in Memory Expansion Mode with 16-bit Bus Address/Data Separate Mode

Unused pins require handling in the circuit (input pins are connected to VDD/VSS, output pins leave open, input/output pins are connected to VDD/VSS or leave open depending on pin direction setting).

1-4-4 Memory Expansion Mode with 8-bit Bus Address/Data Shared Mode

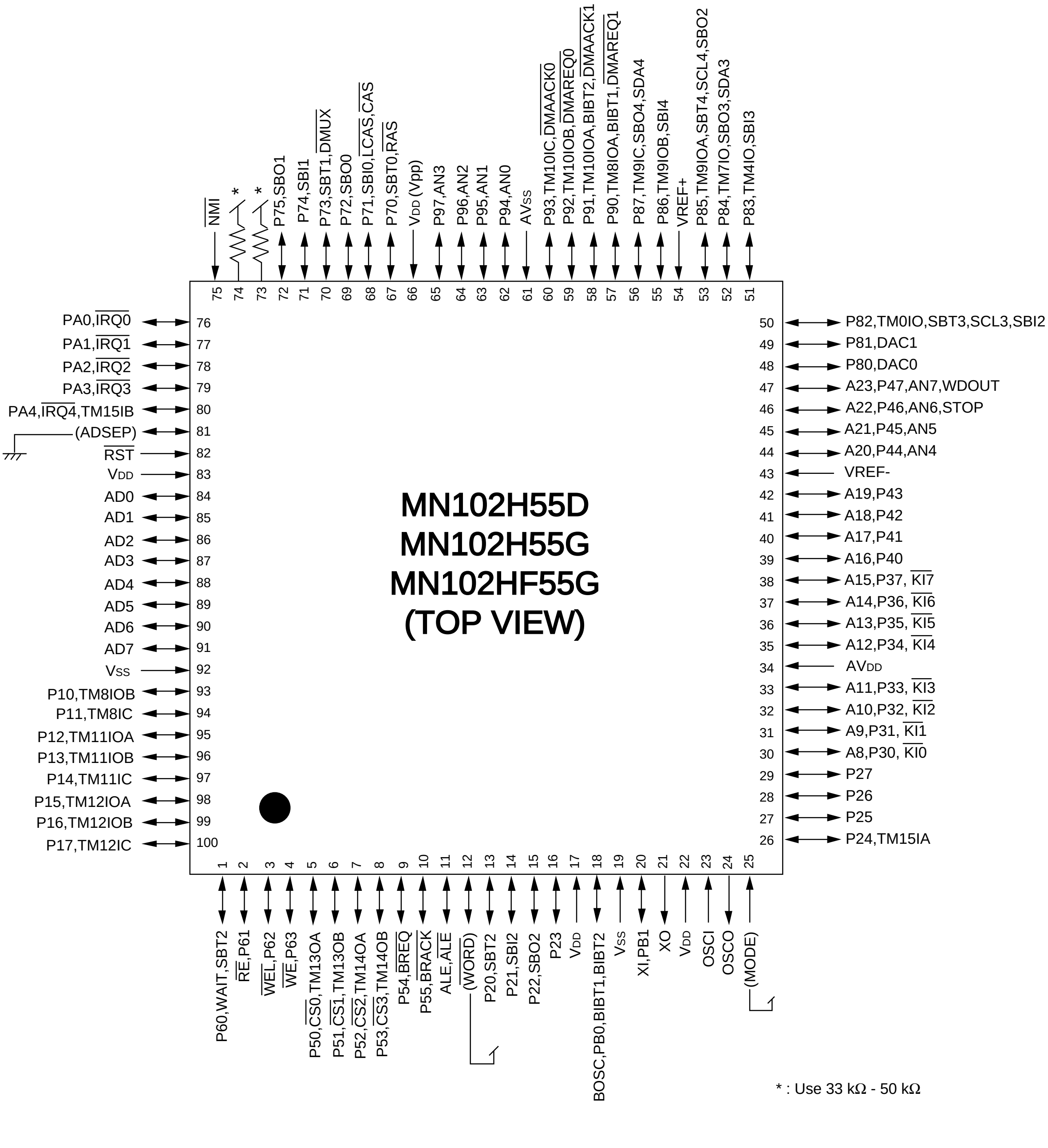


Figure 1-4-4 Pin Configuration in Memory Expansion Mode with 8-bit Bus Address/Data Shared Mode

Unused pins require handling in the circuit (input pins are connected to VDD/VSS, output pins leave open, input/output pins are connected to VDD/VSS or leave open depending on pin direction setting).

1-4-5 Memory Expansion Mode with 16-bit Bus Address/Data Shared Mode

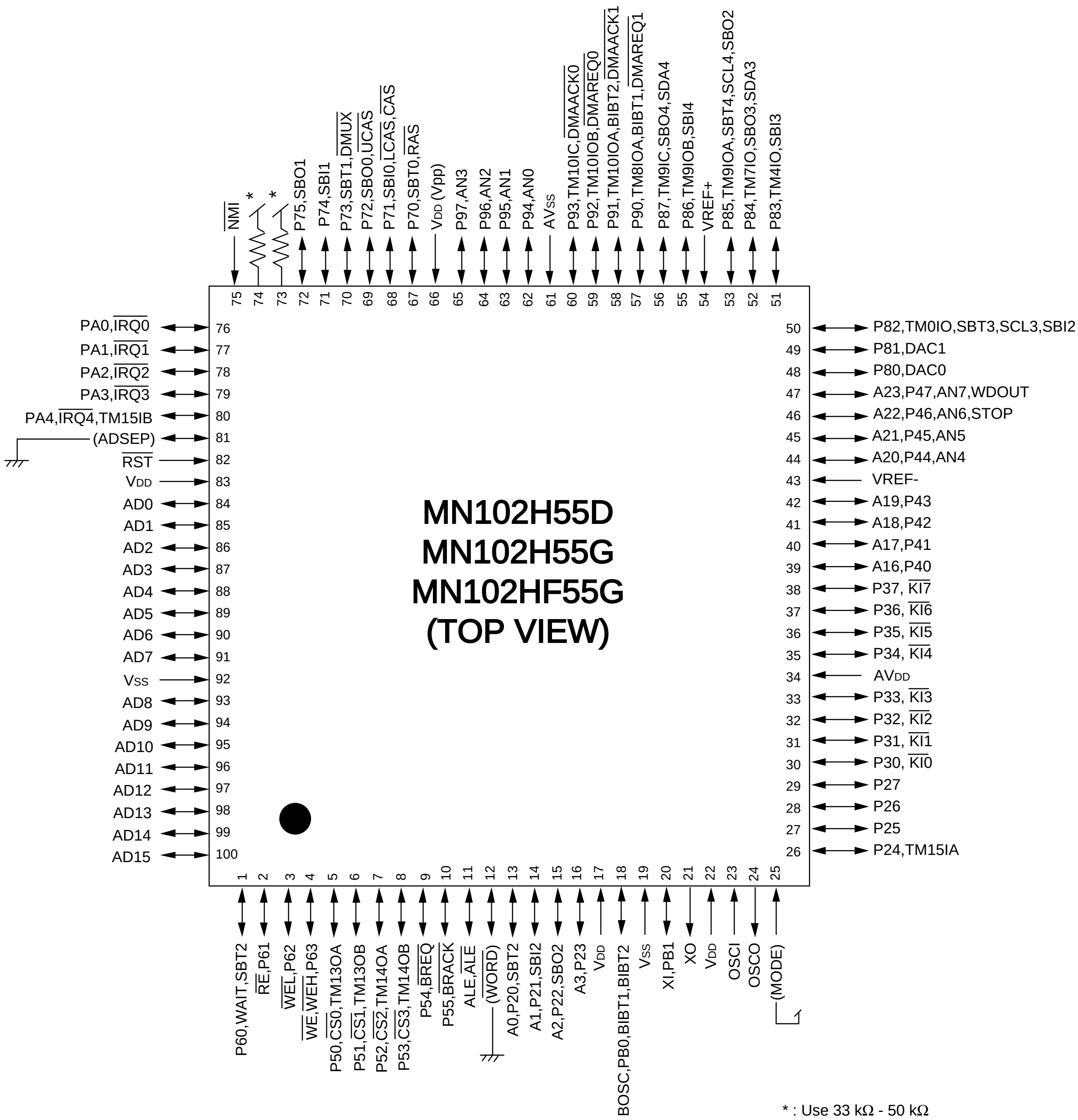


Figure 1-4-5 Pin Configuration in Memory Expansion Mode with 16-bit Bus Address/Data Shared Mode

Unused pins require handling in the circuit (input pins are connected to VDD/VSS, output pins leave open, input/output pins are connected to VDD/VSS or leave open depending on pin direction setting).

1-4-6 Processor Mode with 8-bit Bus Address/Data Separate Mode

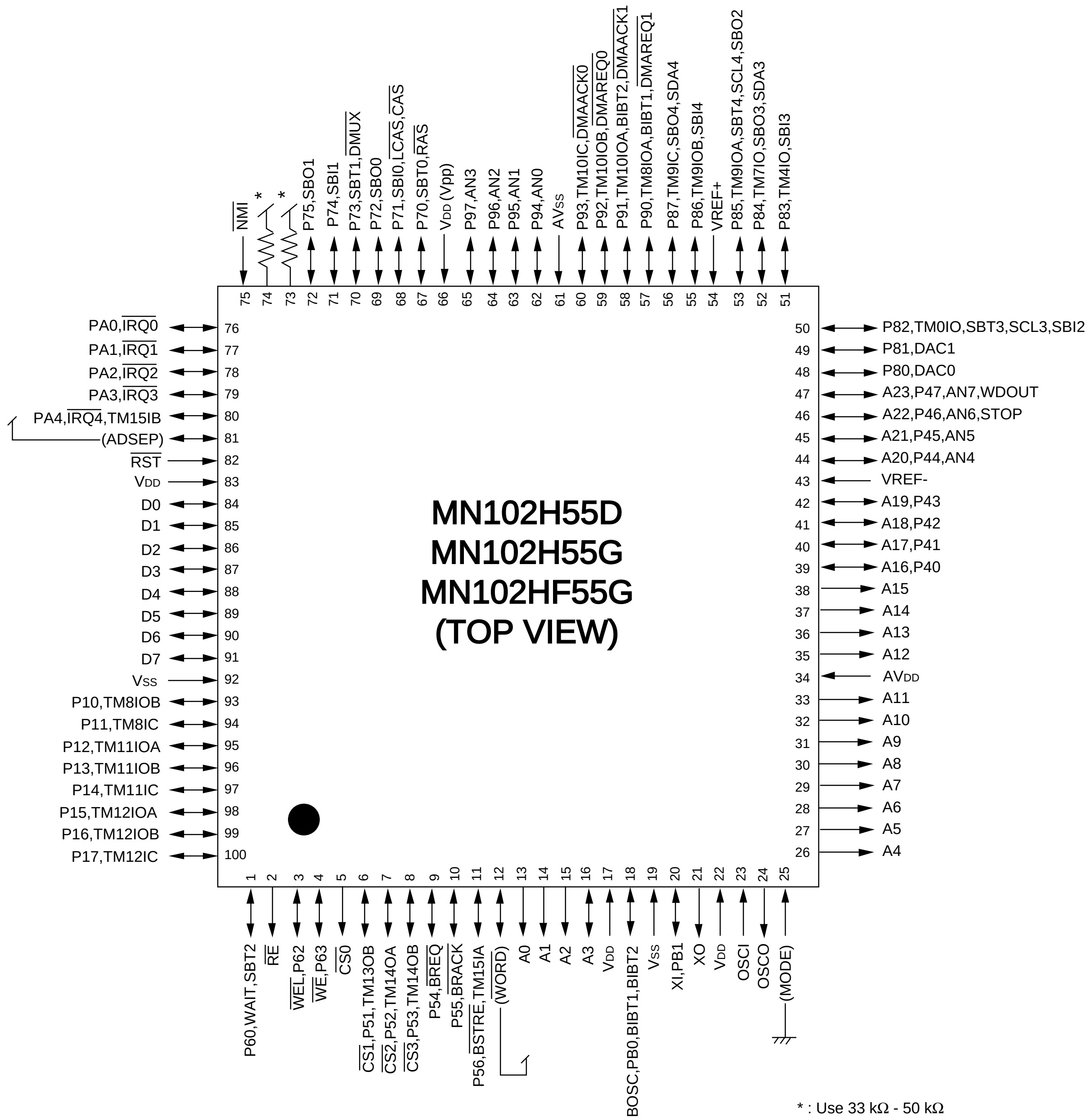


Figure 1-4-6 Pin Configuration in Processor Mode with 8-bit Bus Address/Data Separate Mode

Unused pins require handling in the circuit (input pins are connected to VDD/VSS, output pins leave open, input/output pins are connected to VDD/VSS or leave open depending on pin direction setting).

1-4-7 Processor Mode with 16-bit Bus Address/Data Separate Mode

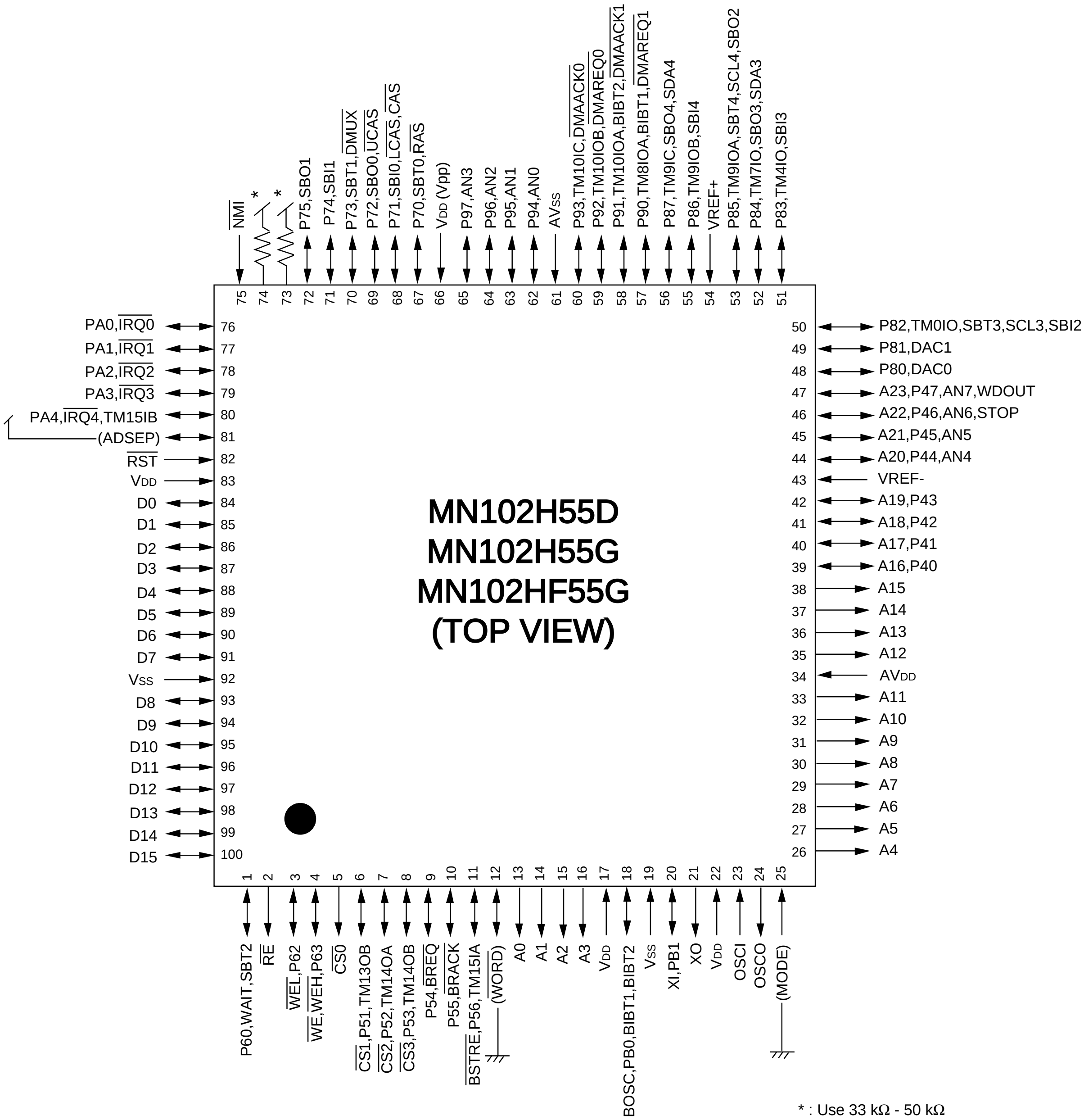


Figure 1-4-7 Pin Configuration in Processor Mode with 16-bit Bus Address/Data Separate Mode

> **!** Unused pins require handling in the circuit (input pins are connected to VDD/VSS, output pins leave open, input/output pins are connected to VDD/VSS or leave open depending on pin direction setting).

1-4-8 Processor Mode with 8-bit Bus Address/Data Shared Mode

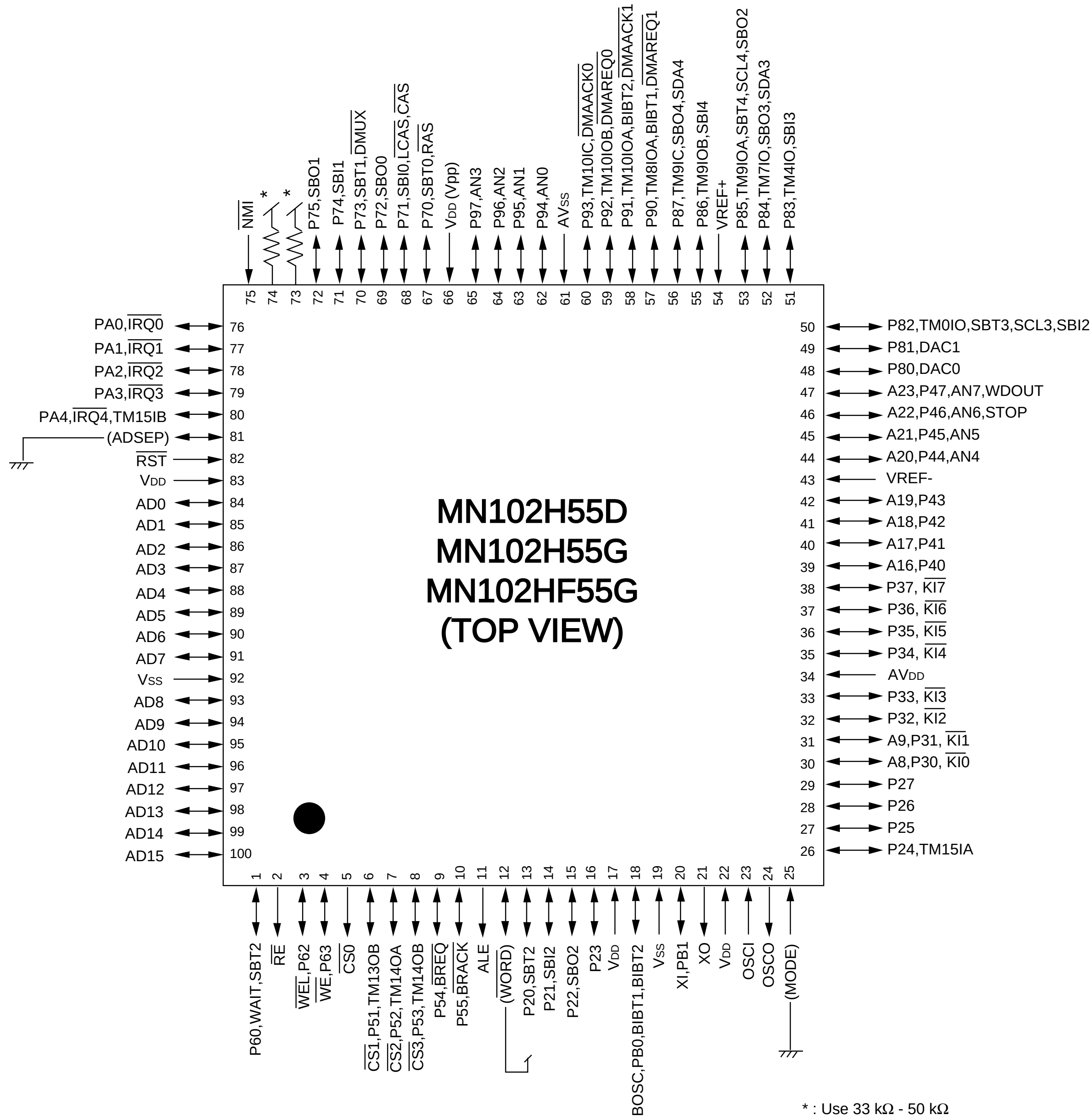


Figure 1-4-8 Pin Configuration in Processor Mode with 8-bit Bus Address/Data Shared Mode

Unused pins require handling in the circuit (input pins are connected to VDD/VSS, output pins leave open, input/output pins are connected to VDD/VSS or leave open depending on pin direction setting).

1-4-9 Processor Mode with 16-bit Bus Address/Data Shared Mode

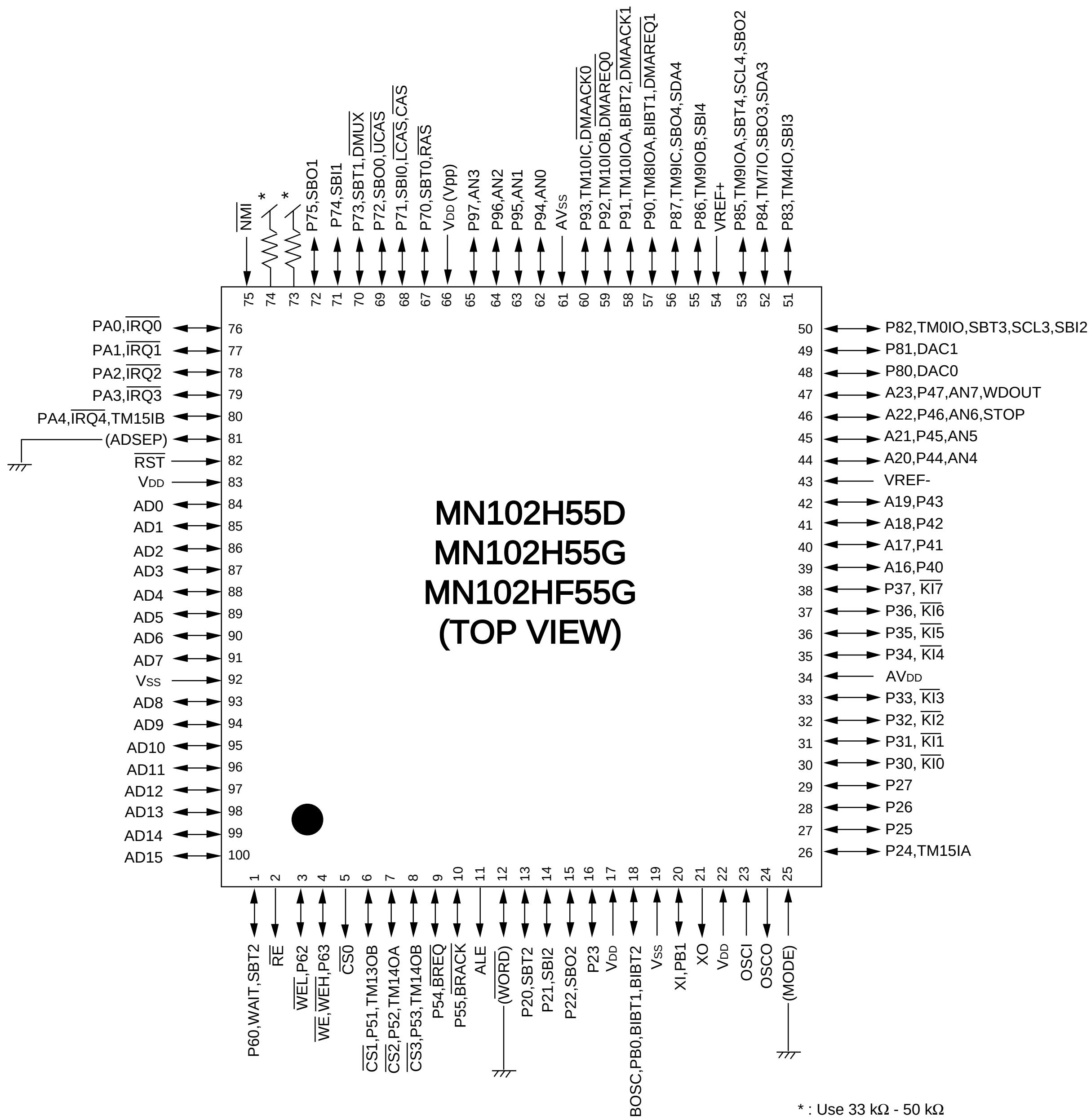


Figure 1-4-9 Pin Configuration in Processor Mode with 16-bit Bus Address/Data Shared Mode

Unused pins require handling in the circuit (input pins are connected to VDD/VSS, output pins leave open, input/output pins are connected to VDD/VSS or leave open depending on pin direction setting).

1-4-10 List of Pin Functions

Refer to "11-2-3 List of Pin Functions" for each pin's input level and Schmidt availability. TTL in the input level column means that the input is determined at TTL level. CMOS in the input level column means that the input is determined at CMOS level. The column with "yes" sign shows Schmidt, while the column with no mark shows no Schmidt. Pull-up can be programmable with the pull-up control registers. Please refer to "Chapter 8 Ports" for details. The unused pins require handling on the board. The input pins are connected to VDD or VSS. The output pins leave open. The lack of this handling causes the increase of current and unstable operation.

Table 1-4-1 List of Pin Functions (1/26)

Pin Number	Pin Name	I/O	Function	Description
17 22 66 83	V_{DD} V_{DD} V_{DD} (V_{PP}) V_{DD}	- - - -	Power Power Power Power	There are four V_{DD} pins. The mask ROM chip (MN102H55D/55G) must connect these four pins to a power supply of 3.0 V to 3.6 V. The flash ROM chip (MN102HF55G) must connect pin 66 to a power supply of 4.5 V to 5.5 V because the pin 66 becomes the power pin for the flash ROM programming. During the normal operation, the pin 66 must be input the same voltage of other pins.
19 92	V_{SS} V_{SS}	- -	Power (Ground) Power (Ground)	There are two Vss pins. They must be connected to a power supply of 0 V.
34	AV_{DD}	-	Analog Voltage	There is one AV_{DD}. It must be connected to the same voltage as V_{DD}.
61	AV_{SS}	-	Analog Voltage (Ground)	There is one AV_{SS}. It must be connected to the same voltage as V_{SS}.
43	Vref -	-	Analog Basic Voltage	There is one Vref -. It must be connected with relation of V_{SS} <= Vref - < Vref + <= V_{DD}.
54	Vref +	-	Analog Basic Voltage	There is one Vref +. It must be connected with relation of V_{SS} <= Vref - < Vref + <= V_{DD}.

Table 1-4-1 List of Pin Functions (2/26)

Pin Number	Pin Name	I/O	Function	Description
23 24	OSCI OSCO	Input Output	High-speed Oscillator Input High-speed Oscillator Output	For a self-excited oscillator configuration, connect crystal or ceramic oscillator across these two pins. They have a built-in feedback resistor between them. For stability, insert capacitor of 20 pF to 33 pF between the OSCI pin or the OSCO pin and the V_{SS} pin (For the exact capacitance, consult the oscillator manufacturer). For an external oscillator configuration, connect the OSCI pin to an oscillator with an amplitude of 4 MHz to 34 MHz and the width between V_{DD} and V_{SS}. Leave the OSCO pin open. Refer to "Figure 1-4-10". Connecting the OSCO pin with the external circuit directly is not allowed when the oscillator clock is taken from the chip. Select the BOSC pin for a synchronous signal.
20 21	XI XO	Input Output	Low-speed Oscillator Input Low-speed Oscillator Output	For a self-excited oscillator configuration, connect crystal or ceramic oscillator across these two pins. They have a built-in feedback resistor between them. For stability, insert capacitor of 100 pF to 200 pF between the XI pin or the XO pin and the V_{SS} pin (For the exact capacitance, consult the oscillator manufacturer). For an external oscillator configuration, connect the XI pin to an oscillator with an amplitude of 32 kHz to 166 kHz and the width between V_{DD} and V_{SS}. Leave the XO pin open. Refer to "Figure 1-4-11". If the XI pin is not used as the low-speed oscillator input pin, connect the XI pin to V_{SS} or V_{DD}. If the XO pin is not used as the low-speed oscillator output pin, leave the XO pin open. When the oscillation clock is taken from the chip, connecting the XO pin with the external circuit directly is not allowed. Select the BOSC pin for a synchronous signal.
	PB1	I/O	General-purpose port B1 pin	If pin 20 is not used as the XI pin, this pin can be used as the general-purpose I/O port. The PBMD register switches the function. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".

Table 1-4-1 List of Pin Functions (3/26)

Pin Number	Pin Name	I/O	Function	Description
82	$\overline{RST}$	Input	Reset Input	This pin resets the chip. With a 34-MHz oscillator, reset starts when the low level is input to this pin for more than 117 ns. Reset starts even when the noise is input to this pin for 117 ns. When the high level is input to the pin, reset is released. After the reset pin becomes high level, the oscillation waits of the high-speed oscillation pins (OSCI and OSCO) are performed (approximately 3.855 ms with a 34-MHz oscillator). After that, the chip starts executing the instruction from x'080000'. Refer to "Figure 1-4-12".
18	BOSC	Output	System Clock Output	This pin provides the system clock. After reset release, the pin outputs BOSC. When the high-speed oscillation pin is operating at 34 MHz, the pin outputs the clock of 34 MHz.
	BIBT1 BIBT2	Output Output	Internal System Clock Output	Pin 18 can output BIBT1 or BIBT2 signal of the internal system clock by setting the PBMD register. These signals are inverted signals.
	PB0	I/O	General-purpose Port B0	If pin 18 is not used as the BOSC pin, it can be used as a general-purpose input/output port. The PBMD register switches the function. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
25	MODE	Input	Mode Setup Input	This pin sets either processor mode or single-chip mode (memory expansion mode). Pulling the pin low sets the processor mode. In processor mode, Internal ROM becomes the external memory area, and the chip executes the instruction from x'080000' in memory connected externally. Pulling the pin high sets the single-chip mode (memory expansion mode). The chip executes the instruction from x'080000' of Internal ROM. In memory expansion mode, the port mode register is set to address output and data output by instruction. Do not change the mode setting in this pin during operation. When the setting is changed, proper operation cannot be guaranteed. Refer to "2-1 Summary of Bus Interface".

Table 1-4-1 List of Pin Functions (4/26)

Pin Number	Pin Name	I/O	Function	Description
12	$\overline{\text{WORD}}$	Input	Data Bus Width Setup Input	This pin sets either 8-bit data bus width or 16-bit data bus width in the external memory space 0 immediately after reset release in processor mode or memory expansion mode. Pulling the pin high sets 8-bit bus width while pulling the pin low sets 16-bit bus width. In processor mode or memory expansion mode, this pin must be used as the data bus width setup pin. The MEMMD1 register determines the data bus width for the external memory spaces 1 to 3. The MEMMD1 register can reset the data bus width for the external memory space 0 after reset release, regardless the level of this pin. Refer to "2-1 Summary of Bus Interface".
	P57	I/O	General-purpose Port 57	This pin can be used as a general-purpose input/output port in single-chip mode. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
81	ADSEP	Input	Address/Data Separate, Shared Mode Setup	This pin sets either address/data separate mode or address/data shared mode in processor mode or memory expansion mode. Pulling the pin high sets the address/data separate mode while pulling the pin low sets the address/data shared mode. In processor mode or memory expansion mode, this pin must be used as the address/data separate, shared mode setup pin. Do not change this pin's input during operation. When the setting is changed, proper operation cannot be guaranteed. Refer to "2-1 Summary of Bus Interface".
	PA5	I/O	General-purpose Port A5	This pin can be used as a general-purpose input/output port in single-chip mode. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".

Table 1-4-1 List of Pin Functions (5/26)

Pin Number	Pin Name	I/O	Function	Description
9	$\overline{\text{BREQ}}$	Input	Bus Request Input	$\overline{\text{BREQ}}$ and $\overline{\text{BRACK}}$ pins operate bus arbitration. Pulling $\overline{\text{BREQ}}$ low suspends the execution of the current instruction, makes addresses, data and control signals high impedance, and then releases bus. After that, pull $\overline{\text{BRACK}}$ low. While the chip is accessing the bus, the chip releases the bus after the bus access is completed. Pulling $\overline{\text{BREQ}}$ high at the level detector restores the bus.
	P54	I/O	General-purpose Port 54	This pin can be used as a general-purpose input/output port in single-chip mode. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
10	$\overline{\text{BRACK}}$	Output	Bus Request Enable Output	$\overline{\text{BREQ}}$ and $\overline{\text{BRACK}}$ pins operate bus arbitration. Refer to "Pin 9 $\overline{\text{BREQ}}$ Description" for details.
	P55	I/O	General-purpose Port 55	This pin can be used as a general-purpose input/output port in single-chip mode. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
2	$\overline{\text{RE}}$	Output	Read Enable Output	This pin provides a control signal for the external memory read in processor mode or memory expansion mode. When connecting SRAM and ROM, connect $\overline{\text{RE}}$ to $\overline{\text{OE}}$ in memory. $\overline{\text{RE}}$ outputs low level during read operation and the chip reads the contents of the memory. Refer to "2-1 Summary of Bus Interface". During a bus request, STOP mode or HALT mode, this pin will be in a high impedance state. Refer to "11-2-3 List of Pin Functions".
	P61	I/O	General-purpose Port 61	This pin can be used as a general-purpose input/output port if it is not used as $\overline{\text{RE}}$ in single-chip mode, processor mode or memory expansion mode. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".

Table 1-4-1 List of Pin Functions (6/26)

Pin Number	Pin Name	I/O	Function	Description
3	$\overline{\text{WEL}}$	Output	Lower Byte Write Enable Output	This pin provides a control signal for the external memory write in processor mode or memory expansion mode. When connecting SRAM and ROM, connect $\overline{\text{WEL}}$ to $\overline{\text{WE}}$ in memory. $\overline{\text{WEL}}$ outputs low level when writing the lower bytes (bits 0 to 7 of data) and writes the data to memory. Refer to "2-1 Summary of Bus Interface". During a bus request, STOP mode or HALT mode, this pin will be in a high impedance state. Refer to "11-2-3 List of Pin Functions".
	P62	I/O	General-purpose Port 62	This pin can be used as a general-purpose input/output port if it is not used as $\overline{\text{WEL}}$ in single-chip mode, processor mode or memory expansion mode. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
4	$\overline{\text{WEH}}$	Output	Upper Byte Write Enable Output	This pin provides a control signal for the external memory write in processor mode or memory expansion mode. When connecting SRAM and ROM, connect $\overline{\text{WEH}}$ to $\overline{\text{WE}}$ in memory. $\overline{\text{WEH}}$ outputs low level when writing the upper bytes (bits 8 to 15 of data) and writes the data to memory. $\overline{\text{WEH}}$ is invalid when 8-bit bus width is selected in processor mode or memory expansion mode so that it can be used as a general-purpose port 63 pin. Refer to "2-1 Summary of Bus Interface". During a bus request, STOP mode or HALT mode, this pin will be in a high impedance state. Refer to "11-2-3 List of Pin Functions".
	$\overline{\text{WE}}$	Output	Write Enable Output for DRAM Connection	This pin provides a write enable pin when connecting DRAM in processor mode or memory expansion mode. When connecting DRAM with 2CAS method, connect this pin to $\overline{\text{WE}}$ in DRAM. $\overline{\text{WE}}$ outputs low during write operation and writes the data to DRAM. Refer to "2-1 Summary of Bus Interface". During a bus request, STOP mode or HALT mode, this pin will be in a high impedance state. Refer to "11-2-3 List of Pin Functions".

Table 1-4-1 List of Pin Functions (7/26)

Pin Number	Pin Name	I/O	Function	Description
	P63	I/O	General-purpose Port 63	This pin can be used as a general-purpose input/output port when 8-bit bus width is selected in single-chip mode, processor mode or memory expansion mode. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
5	$\overline{CS0}$	Output	Chip Select Output	This pin provides a chip select signal corresponding to each external memory space when accessing SRAM and ROM connected to the external memory spaces 0 to 3 in processor mode or memory expansion mode. Connect $\overline{CS0}$ - $\overline{CS3}$ to $\overline{CS}$ pins in external memory. $\overline{CS0}$ cannot be output when accessing Internal ROM or Internal RAM. Refer to "2-1 Summary of Bus Interface". During a bus request, STOP mode or HALT mode, this pin will be in a high impedance state. Refer to "11-2-3 List of Pin Functions".
	TM13OA	Output	Timer 13A Output	This pin can be used as a timer 13 PWM output pin if it is not used as a chip select output pin in single-chip mode, processor mode or memory expansion mode. Refer to "Chapter 4 Timers".
6	$\overline{CS1}$	Output	Chip Select Output	Refer to "Pin 5 $\overline{CS0}$ Description" for details.
	TM13OB	Output	Timer 13B Output	This pin can be used as a timer 13 PWM output pin if it is not used as a chip select output pin in single-chip mode, processor mode or memory expansion mode. Refer to "Chapter 4 Timers".
7	$\overline{CS2}$	Output	Chip Select Output	Refer to "Pin 5 $\overline{CS0}$ Description" for details.
	TM14OA	Output	Timer 14A Output	This pin can be used as a timer 14 PWM output pin if it is not used as a chip select output pin in single-chip mode, processor mode or memory expansion mode. Refer to "Chapter 4 Timers".

Table 1-4-1 List of Pin Functions (8/26)

Pin Number	Pin Name	I/O	Function	Description
8	$\overline{CS3}$	Output	Chip Select Output	Refer to "Pin 5 $\overline{CS0}$ Description" for details.
	TM14OB	Output	Timer 14B Output	This pin can be used as a timer 14 PWM output pin if it is not used as a chip select output pin in single-chip mode, processor mode or memory expansion mode. Refer to "Chapter 4 Timers".
11	ALE	Output	Address Latch Enable Output (Positive Logic)	This pin provides a timing signal of latching the address which outputs to AD0 to AD15 pins during address/data shared mode in processor mode or memory expansion mode. ALE outputs at positive logic at reset release, but the P5HMD changes to negative logic. Because of this, ALE cannot be used at negative logic in processor mode. Refer to "2-1 Summary of Bus Interface". During a bus request, STOP mode or HALT mode, this pin will be in a high impedance state.
	$\overline{ALE}$	Output	Address Latch Enable Output (Negative Logic)	
	$\overline{BSTRE}$	Output	Read Enable for Burst ROM	When connecting burst ROM to the external memory space in processor mode or memory expansion mode, connect this pin to $\overline{RE}$ in burst ROM. Refer to "2-1 Summary of Bus Interface". During a bus request, STOP mode or HALT mode, this pin will be in a high impedance state.
	P56	I/O	General-purpose Port 56	This pin can be used as a general-purpose input/output port if it is not used as ALE, $\overline{ALE}$ or $\overline{BSTRE}$ in single-chip mode, processor mode or memory expansion mode. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	TM15IA	Input	Timer 15A Input	This pin can be used as a timer 15 pulse input pin if it is not used as ALE, $\overline{ALE}$ or $\overline{BSTRE}$ in single-chip mode, processor mode or memory expansion mode. Because pin 26 has the same function, either pin 26 or pin 11 must be selected. Refer to "Chapter 4 Timers".

Table 1-4-1 List of Pin Functions (9/26)

Pin Number	Pin Name	I/O	Function	Description
1	WAIT	Input	Bus Cycle Wait Input	This pin extends or shortens the cycle of accessing to the external memory based on the signal inputted to this pin when the external memory wait is set to the handshake mode in processor mode or memory expansion mode. Pulling this pin low ends access to the external memory. Refer to "Figure 1-4-13, Table 2-1-3 to Table 2-1-6".
	P60	I/O	General-purpose Port 60	This pin can be used as a general-purpose input/output port if it is not used as WAIT in single-chip mode, processor mode or memory expansion mode. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	SBT2	I/O	Serial Interface 2 Clock Input/Output	This pin can be used as a synchronous transfer clock signal input/output pin for serial interface 2 if it is not used as WAIT in single-chip mode, processor mode or memory expansion mode. Because pin 13 has the same function, either pin 13 or pin 1 must be selected. Refer to "Chapter 5 Serial Interface".
13	A0	Output	Address Output	This pin outputs the address of the external memory in processor mode or memory expansion mode. Connect this pin to address pin of the external memory or address decode circuit. When it is not accessing the external memory, the output value is undefined. Refer to "2-1 Summary of Bus Interface". During a bus request, STOP mode or HALT mode, this pin will be in a high impedance state.
	P20	I/O	General-purpose Port 20	This pin can be used as a general-purpose input/output port if it is not used as the address output pin in single-chip mode, processor mode or memory expansion mode. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	SBT2	I/O	Serial Interface 2 Clock Input/Output	This pin can be used as a synchronous transfer clock signal input/output pin for serial interface 2 if it is not used as WAIT in single-chip mode, processor mode or memory expansion mode. Because pin 13 has the same function, either pin 13 or pin 1 must be selected. Refer to "Chapter 5 Serial Interface".

Table 1-4-1 List of Pin Functions (10/26)

Pin Number	Pin Name	I/O	Function	Description
14	A1	Output	Address Output	Refer to "Pin 13 A0 Description" for details.
	P21	I/O	General-purpose Port 21	Refer to "Pin 13 P20 Description" for details.
	SBI2	Input	Serial Interface 2 Data Input	This pin can be used as a data input pin for serial interface 2 if it is not used as the address output pin in single-chip mode, processor mode or memory expansion mode. Because pin 50 has the same function, either pin 50 or pin 14 must be selected. Refer to "Chapter 5 Serial Interface".
15	A2	Output	Address Output	Refer to "Pin 13 A0 Description" for details.
	P22	I/O	General-purpose Port 21	Refer to "Pin 13 P20 Description" for details.
	SBO2	Output	Serial Interface 2 Data Output	This pin can be used as a data output pin for serial interface 2 if it is not used as the address output pin in single-chip mode, processor mode or memory expansion mode. Because pin 53 has the same function, either pin 53 or pin 15 must be selected. Refer to "Chapter 5 Serial Interface".
16	A3	Output	Address Output	Refer to "Pin 13 A0 Description" for details.
	P23	I/O	General-purpose Port 23	Refer to "Pin 13 P20 Description" for details.

Table 1-4-1 List of Pin Functions (11/26)

Pin Number	Pin Name	I/O	Function	Description
26	A4	Output	Address Output	Refer to "Pin 13 A0 Description" for details.
	P24	I/O	General-purpose Port 24	Refer to "Pin 13 P20 Description" for details.
	TM15IA	Input	Timer 15 Pulse Input	This pin can be used as a timer 15 pulse input pin if it is not used as the address output pin in single-chip mode, processor mode or memory expansion mode. Because pin 11 has the same function, either pin 11 or pin 26 must be selected. Refer to "Chapter 4 Timers".
27-29	A5-A7	Output	Address Output	Refer to "Pin 13 A0 Description" for details.
	P25-P27	I/O	General-purpose Port 25-27	Refer to "Pin 13 P20 Description" for details.
30-33, 35-38	A8-A15	Output	Address Output	Refer to "Pin 13 A0 Description" for details.
	P30-P37	I/O	General-purpose Port 30-37	Refer to "Pin 13 P20 Description" for details.
	$\overline{KI0}$-$\overline{KI7}$	Input	Key Input Interrupt	These pins can be used as key input interrupt pins if they are not used as the address output pins in single-chip mode, processor mode or memory expansion mode. The key input interrupt pins can be controlled in bit units. Refer to "3-2-2 Key Input Interrupt Setup Examples".

Table 1-4-1 List of Pin Functions (12/26)

Pin Number	Pin Name	I/O	Function	Description
39-42	A16-A19	Output	Address Output	Refer to "Pin 13 A0 Description" for details.
	P40-P43	I/O	General-purpose Port 40-43	Refer to "Pin 13 P20 Description" for details.
44-45	A20-A21	Output	Address Output	Refer to "Pin 13 A0 Description" for details.
	P44-P45	I/O	General-purpose Port 44-45	Refer to "Pin 13 P20 Description" for details.
	AN4-AN5	Input	A/D Conversion Input	These pins can be used as A/D conversion input pins if they are not used as address output pins in single-chip mode, processor mode or memory expansion mode. Refer to "6-1 Summary of A/D Converter".
46	A22	Output	Address Output	Refer to "Pin 13 A0 Description" for details.
	P46	I/O	General-purpose Port 46	Refer to "Pin 13 P20 Description" for details.
	AN6	Input	A/D Converter 6 Conversion Input	This pin can be used as a A/D conversion input pin if it is not used as the address output pin in single-chip mode, processor mode or memory expansion mode. Refer to "6-1 Summary of A/D Converter".
	STOP	Output	STOP Status Output	This pin outputs high to indicate STOP status if it is not used as the address output pin in single-chip mode, processor mode or memory expansion mode.

Table 1-4-1 List of Pin Functions (13/26)

Pin Number	Pin Name	I/O	Function	Description
47	A23	Output	Address Output	Refer to "Pin 13 A0 Description" for details.
	P47	I/O	General-purpose Port 47	Refer to "Pin 13 P20 Description" for details.
	AN7	Input	A/D Converter 7 Conversion Input	This pin can be used as a A/D conversion input pin if it is not used as the address output pin in single-chip mode, processor mode or memory expansion mode. Refer to "6-1 Summary of A/D Converter".
	WDOUT	Output	Watchdog Timer Overflow Output	This pin outputs high when the watchdog timer overflows if it is not used as the address output pin in single-chip mode, processor mode or memory expansion mode.
84-91	D0-D7 AD0-AD7	I/O I/O	Data I/O Address/Data I/O	These pins input or output the lower 8-bit data of the external memory during address/data separate mode in processor mode or memory expansion mode. During address/data shared mode, these pins time-divides input or output the lower 8-bit address and the lower 8-bit data of the external memory. They become input when the external memory is not accessed. Refer to "2-1 Summary of Bus Interface". During a bus request, STOP mode or HALT mode, this pin will be in a high impedance state.
	P00-P07	I/O	General-purpose Ports 00-07	These pins can be used as general-purpose ports if they are not used as data input/output pins or address/data input/output pins in single-chip mode, processor mode or memory expansion mode. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".

Table 1-4-1 List of Pin Functions (14/26)

Pin Number	Pin Name	I/O	Function	Description
93	D8 AD8	I/O I/O	Data I/O Address/Data I/O	This pin inputs or outputs the upper 8-bit data of the external memory during address/data separate mode in processor mode or memory expansion mode. During address/data shared mode, this pin time-divides input or output the upper 8-bit address and the upper 8-bit data of the external memory. It becomes input when the external memory is not accessed. Refer to "2-1 Summary of Bus Interface". During a bus request, STOP mode or HALT mode, this pin will be in a high impedance state.
	P10	I/O	General-purpose Port 10	Refer to "Pins 84-91 P00-P07 Description" for details.
	TM8IOB	I/O	Timer 8B Input/Output	This pin can be used as a timer 8 input capture B input pin or a timer 8 output compare B output pin if it is not used as a data input/output pin or an address/data input/output pin in single-chip mode, processor mode or memory expansion mode. Refer to "Chapter 4 Timers".
94	D9 AD9	I/O I/O	Data I/O Address/Data I/O	Refer to "Pin 93 D8, AD8 Description" for details.
	P11	I/O	General-purpose Port 11	Refer to "Pins 84-91 P00-P07 Description" for details.
	TM8IC	Input	Timer 8C Input	This pin can be used as a timer 8 counter clear pin if it is not used as a data input/output pin or an address/data input/output pin in single-chip mode, processor mode or memory expansion mode. Refer to "Chapter 4 Timers".

Table 1-4-1 List of Pin Functions (15/26)

Pin Number	Pin Name	I/O	Function	Description
95	D10 AD10	I/O I/O	Data I/O Address/Data I/O	Refer to "Pin 93 D8, AD8 Description" for details.
	P12	I/O	General-purpose Port 12	Refer to "Pins 84-91 P00-P07 Description" for details.
	TM11IOA	I/O	Timer 11A Input/Output	This pin can be used as a timer 11 input capture A input pin or a timer 11 output compare A output pin if it is not used as a data input/output pin or an address/data input/output pin in single-chip mode, processor mode or memory expansion mode. Refer to "Chapter 4 Timers".
96	D11 AD11	I/O I/O	Data I/O Address/Data I/O	Refer to "Pin 93 D8, AD8 Description" for details.
	P13	I/O	General-purpose Port 13	Refer to "Pins 84-91 P00-P07 Description" for details.
	TM11IOB	I/O	Timer 11B Input/Output	This pin can be used as a timer 11 input capture B input pin or a timer 11 output compare B output pin if it is not used as a data input/output pin or an address/data input/output pin in single-chip mode, processor mode or memory expansion mode. Refer to "Chapter 4 Timers".
97	D12 AD12	I/O I/O	Data I/O Address/Data I/O	Refer to "Pin 93 D8, AD8 Description" for details.
	P14	I/O	General-purpose Port 14	Refer to "Pins 84-91 P00-P07 Description" for details.
	TM11IC	Input	Timer 11C Input	This pin can be used as a timer 11 counter clear pin if it is not used as a data input/output pin or an address/data input/output pin in single-chip mode, processor mode or memory expansion mode. Refer to "Chapter 4 Timers".

Table 1-4-1 List of Pin Functions (16/26)

Pin Number	Pin Name	I/O	Function	Description
98	D13 AD13	I/O I/O	Data I/O Address/Data I/O	Refer to "Pin 93 D8, AD8 Description" for details.
	P15	I/O	General-purpose Port 15	Refer to "Pins 84-91 P00-P07 Description" for details.
	TM12IOA	I/O	Timer 12A Input/Output	This pin can be used as a timer 12 input capture A input pin or a timer 12 output compare A output pin if it is not used as a data input/output pin or an address/data input/output pin in single-chip mode, processor mode or memory expansion mode. Refer to "Chapter 4 Timers".
99	D14 AD14	I/O I/O	Data I/O Address/Data I/O	Refer to "Pin 93 D8, AD8 Description" for details.
	P16	I/O	General-purpose Port 16	Refer to "Pins 84-91 P00-P07 Description" for details.
	TM12IOB	I/O	Timer 12B Input/Output	This pin can be used as a timer 12 input capture B input pin or a timer 12 output compare B output pin if it is not used as a data input/output pin or an address/data input/output pin in single-chip mode, processor mode or memory expansion mode. Refer to "Chapter 4 Timers".
100	D15 AD15	I/O I/O	Data I/O Address/Data I/O	Refer to "Pin 93 D8, AD8 Description" for details.
	P17	I/O	General-purpose Port 17	Refer to "Pins 84-91 P00-P07 Description" for details.
	TM12IC	Input	Timer 12C Input	This pin can be used as a timer 12 counter clear pin if it is not used as a data input/output pin or an address/data input/output pin in single-chip mode, processor mode or memory expansion mode. Refer to "Chapter 4 Timers".

Table 1-4-1 List of Pin Functions (17/26)

Pin Number	Pin Name	I/O	Function	Description
67	$\overline{\text{RAS}}$	Output	DRAM Control Output	This pin outputs $\overline{\text{RAS}}$ signal when connecting DRAM in processor mode or memory expansion mode. Refer to "2-1 Summary of Bus Interface". During a bus request, STOP mode or HALT mode, this pin will be in a high impedance state.
	P70	I/O	General-purpose Port 70	This pin can be used as a general-purpose input/output port if it is not used as $\overline{\text{RAS}}$ in single-chip mode, processor mode or memory expansion mode. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	SBT0	I/O	Serial Interface 0 Clock Input/Output	This pin can be used as a synchronous transfer clock signal input/output pin for serial interface 0 if it is not used as $\overline{\text{RAS}}$ in single-chip mode, processor mode or memory expansion mode. Refer to "Chapter 5 Serial Interface".
68	$\overline{\text{CAS}}$ $\overline{\text{LCAS}}$	Output Output	DRAM Control Output DRAM Control Output	This pin outputs $\overline{\text{CAS}}$ or $\overline{\text{LCAS}}$ signal when connecting DRAM in processor mode or memory expansion mode. Refer to "2-1 Summary of Bus Interface". During a bus request, STOP mode or HALT mode, this pin will be in a high impedance state.
	P71	I/O	General-purpose Port 71	This pin can be used as a general-purpose input/output port if it is not used as $\overline{\text{CAS}}$ or $\overline{\text{LCAS}}$ in single-chip mode, processor mode or memory expansion mode. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	SBI0	Input	Serial Interface 0 Data Input	This pin can be used as a data input pin for serial interface 0 if it is not used as $\overline{\text{CAS}}$ or $\overline{\text{LCAS}}$ in single-chip mode, processor mode or memory expansion mode. Refer to "Chapter 5 Serial Interface".

Table 1-4-1 List of Pin Functions (18/26)

Pin Number	Pin Name	I/O	Function	Description
69	$\overline{UCAS}$	Output	DRAM Control Output	This pin outputs $\overline{UCAS}$ signal when connecting DRAM in processor mode or memory expansion mode. Refer to "2-1 Summary of Bus Interface". During a bus request, STOP mode or HALT mode, this pin will be in a high impedance state.
	P72	I/O	General-purpose Port 72	This pin can be used as a general-purpose input/output port if it is not used as $\overline{UCAS}$ in single-chip mode, processor mode or memory expansion mode. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	SBO0	Output	Serial Interface 0 Data Output	This pin can be used as a data output pin for serial interface 0 if it is not used as $\overline{UCAS}$ in single-chip mode, processor mode or memory expansion mode. Refer to "Chapter 5 Serial Interface".
70	$\overline{DMUX}$	Output	DRAM Control Output	This pin outputs $\overline{DMUX}$ signal when connecting DRAM in processor mode or memory expansion mode. Refer to "2-1 Summary of Bus Interface". During a bus request, STOP mode or HALT mode, this pin will be in a high impedance state.
	P73	I/O	General-purpose Port 73	This pin can be used as a general-purpose input/output port if it is not used as $\overline{DMUX}$ in single-chip mode, processor mode or memory expansion mode. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	SBT1	I/O	Serial Interface 1 Clock Input/Output	This pin can be used as a synchronous transfer clock signal input/output pin for serial interface 1 if it is not used as $\overline{DMUX}$ in single-chip mode, processor mode or memory expansion mode. Refer to "Chapter 5 Serial Interface".

Table 1-4-1 List of Pin Functions (19/26)

Pin Number	Pin Name	I/O	Function	Description
71	P74	I/O	General-purpose Port 74	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	SBI1	Input	Serial Interface 1 Data Input	This pin can be used as a data input pin for serial interface 1. Refer to "Chapter 5 Serial Interface".
72	P75	I/O	General-purpose Port 75	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	SBO1	Output	Serial Interface 1 Data Output	This pin can be used as a data output pin for serial interface 1. Refer to "Chapter 5 Serial Interface".
48	P80	I/O	General-purpose Port 80	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	DAC0	Output	D/A Converter 0 Conversion Output	This pin can be used as a pin to output the D/A conversion results. Refer to "6-3 Summary of D/A Converter".
49	P81	I/O	General-purpose Port 81	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	DAC1	Output	D/A Converter 1 Conversion Output	This pin can be used as a pin to output the D/A conversion results. Refer to "6-3 Summary of D/A Converter".

Table 1-4-1 List of Pin Functions (20/26)

Pin Number	Pin Name	I/O	Function	Description
50	P82	I/O	General-purpose Port 82	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	TM0IO	I/O	Timer 0 Input/Output	This pin can be used as a timer 0 input/output pin. Refer to "Chapter 4 Timers".
	SBI2	Input	Serial Interface 2 Data Input	This pin can be used as a data input pin for serial interface 2. Because pin 14 has the same function, either pin 14 or pin 50 must be selected. Refer to "Chapter 5 Serial Interface".
	SBT3	I/O	Serial Interface 3 Clock Input/Output	This pin can be used as a synchronous transfer clock signal input/output pin for serial interface 3. Refer to "Chapter 5 Serial Interface".
	SCL3	Output	Serial Interface 3 Clock Output	This pin can be used as an I^2C clock signal output pin for serial interface 3. Refer to "Chapter 5 Serial Interface".
51	P83	I/O	General-purpose Port 83	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	TM4IO	I/O	Timer 4 Input/Output	This pin can be used as a timer 4 input/output pin. Refer to "Chapter 4 Timers".
	SBI3	Input	Serial Interface 3 Data Input	This pin can be used as a data input pin for serial interface 3. Refer to "Chapter 5 Serial Interface".

Table 1-4-1 List of Pin Functions (21/26)

Pin Number	Pin Name	I/O	Function	Description
52	P84	I/O	General-purpose Port 84	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	TM7IO	I/O	Timer 7 Input/Output	This pin can be used as a timer 7 input/output pin. Refer to "Chapter 4 Timers".
	SBO3	Output	Serial Interface 3 Data Output	This pin can be used as a data output pin for serial interface 3. Refer to "Chapter 5 Serial Interface".
	SDA3	I/O	Serial Interface 3 Data Input/Output	This pin can be used as an I^2C data input/output pin for serial interface 3. Refer to "Chapter 5 Serial Interface".
53	P85	I/O	General-purpose Port 85	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	TM9IOA	I/O	Timer 9A Input/Output	This pin can be used as a timer 9 input/output pin. Refer to "Chapter 4 Timers".
	SBO2	Output	Serial Interface 2 Data Output	This pin can be used as a data output pin for serial interface 2. Because pin 15 has the same function, either pin 15 or pin 53 must be selected. Refer to "Chapter 5 Serial Interface".
	SBT4	I/O	Serial Interface 4 Clock Input/Output	This pin can be used as a synchronous transfer clock signal input/output pin for serial interface 4. Refer to "Chapter 5 Serial Interface".
	SCL4	Output	Serial Interface 4 Clock Output	This pin can be used as an I^2C clock signal output pin for serial interface 4. Refer to "Chapter 5 Serial Interface".

Table 1-4-1 List of Pin Functions (22/26)

Pin Number	Pin Name	I/O	Function	Description
55	P86	I/O	General-purpose Port 86	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	TM9IOB	I/O	Timer 9B Input/Output	This pin can be used as a timer 9 input/output pin. Refer to "Chapter 4 Timers".
	SBI4	Input	Serial Interface 4 Data Input	This pin can be used as a data input pin for serial interface 4. Refer to "Chapter 5 Serial Interface".
56	P87	I/O	General-purpose Port 87	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	TM9IC	Input	Timer 9C Input	This pin can be used as a timer 9 count clear input pin. Refer to "Chapter 4 Timers".
	SBO4	Output	Serial Interface 4 Data Output	This pin can be used as a data output pin for serial interface 4. Refer to "Chapter 5 Serial Interface".
	SDA4	I/O	Serial Interface 4 Data Input/Output	This pin can be used as an I^2C data input/output pin for serial interface 4. Refer to "Chapter 5 Serial Interface".

Table 1-4-1 List of Pin Functions (23/26)

Pin Number	Pin Name	I/O	Function	Description
57	P90	I/O	General-purpose Port 90	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	TM8IOA	I/O	Timer 8A Input/Output	This pin can be used as a timer 8 input/output pin. Refer to "Chapter 4 Timers".
	BIBT1	Output	Internal System Clock Output	Refer to "Pin 18 BIBT1 Description" for details. Refer to "Chapter 5 Serial Interface".
	$\overline{\text{DMAREQ1}}$	Input	ETC1 Activation Request Input	This pin is an ETC activation request pin. When ETC starts, the data is transferred automatically between the external memory and the external device which requires no address specification. Refer to "Chapter 7 ATC, ETC".
58	P91	I/O	General-purpose Port 91	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	TM10IOA	I/O	Timer 10A Input/Output	This pin can be used as a timer 10 input/output pin. Refer to "Chapter 4 Timers".
	BIBT2	Output	Internal System Clock Output	Refer to "Pin 18 BIBT1 Description" for details. Refer to "Chapter 5 Serial Interface".
	$\overline{\text{DMAACK1}}$	Output	ETC1 Acknowledge Output	This pin is an acknowledge signal output pin for ETC activation request. Refer to "Chapter 7 ATC, ETC".

Table 1-4-1 List of Pin Functions (24/26)

Pin Number	Pin Name	I/O	Function	Description
59	P92	I/O	General-purpose Port 92	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	TM10IOB	I/O	Timer 10B Input/Output	This pin can be used as a timer 10 input/output pin. Refer to "Chapter 4 Timers".
	$\overline{\text{DMAREQ0}}$	Input	ETC0 Activation Request Input	This pin is an ETC activation request pin. When ETC starts, the data is transferred automatically between the external memory and the external device which requires no address specification. Refer to "Chapter 7 ATC, ETC".
60	P93	I/O	General-purpose Port 93	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	TM10IC	Input	Timer 10C Input	This pin can be used as a timer 10 counter clear input pin. Refer to "Chapter 4 Timers".
	$\overline{\text{DMAACK0}}$	Output	ETC0 Acknowledge Output	This pin is an acknowledge signal output pin for ETC activation request. Refer to "Chapter 7 ATC, ETC".
62	P94	I/O	General-purpose Port 94	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	AN0	Input	A/D Converter 0 Conversion Input	This pin can be used as an A/D conversion input pin. Refer to "6-1 Summary of A/D Converter".

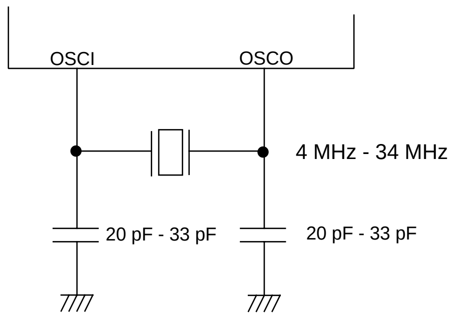
Table 1-4-1 List of Pin Functions (25/26)

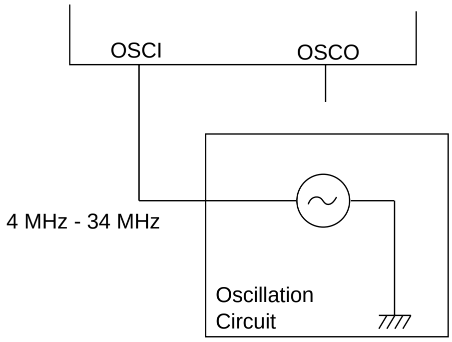
Pin Number	Pin Name	I/O	Function	Description
63	P95	I/O	General-purpose Port 95	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	AN1	Input	A/D Converter 1 Conversion Input	This pin can be used as an A/D conversion input pin. Refer to "6-1 Summary of A/D Converter".
64	P96	I/O	General-purpose Port 96	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	AN2	Input	A/D Converter 2 Conversion Input	This pin can be used as an A/D conversion input pin. Refer to "6-1 Summary of A/D Converter".
65	P97	I/O	General-purpose Port 97	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	AN3	Input	A/D Converter 3 Conversion Input	This pin can be used as an A/D conversion input pin. Refer to "6-1 Summary of A/D Converter".
76	PA0	I/O	General-purpose Port A0	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	$\overline{\text{IRQ0}}$	Input	External Interrupt 0 Input	This pin can be used as an external interrupt request input pin. Refer to "Chapter 3 Interrupts".
77	PA1	I/O	General-purpose Port A1	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	$\overline{\text{IRQ1}}$	Input	External Interrupt 1 Input	This pin can be used as an external interrupt request input pin. Refer to "Chapter 3 Interrupts".

Table 1-4-1 List of Pin Functions (26/26)

Pin Number	Pin Name	I/O	Function	Description
78	PA2	I/O	General-purpose Port A2	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	$\overline{\text{IRQ2}}$	Input	External Interrupt 2 Input	This pin can be used as an external interrupt request input pin. Refer to "Chapter 3 Interrupts".
79	PA3	I/O	General-purpose Port A3	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	$\overline{\text{IRQ3}}$	Input	External Interrupt 3 Input	This pin can be used as an external interrupt request input pin. Refer to "Chapter 3 Interrupts".
80	PA4	I/O	General-purpose Port A4	This pin can be used as a general-purpose input/output port. The input/output direction is controlled in bit units. The pin has a built-in software control pull-up resistor. Refer to "Chapter 8 Ports".
	$\overline{\text{IRQ4}}$	Input	External Interrupt 4 Input	This pin can be used as an external interrupt request input pin. Refer to "Chapter 3 Interrupts".
	TM15IB	Input	Timer 15B Input	This pin can be used as a base clock input pin for timer 15 pulse width measurement. Refer to "Chapter 4 Timers".
75	$\overline{\text{NMI}}$	Input	Nonmaskable Interrupt Input	This pin can be used as a $\overline{\text{NMI}}$ interrupt pin. The /NMI interrupt occurs on the falling edge of low level. In addition, this pin can reads the pin state as the general-purpose input port P76. Refer to "Chapter 3 Interrupts".
73-74	PULLUP	Input	Pull-up	These pins must be pullde up with 33 kΩ - 50 kΩ.

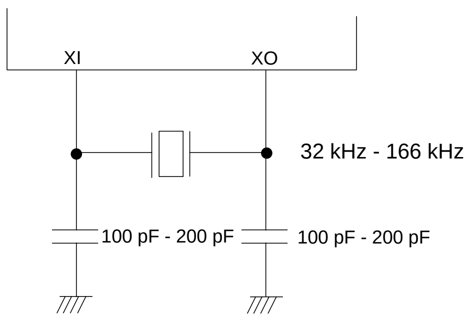
■ Connection Examples of Power Pins, Oscillator Circuits, Reset Pins

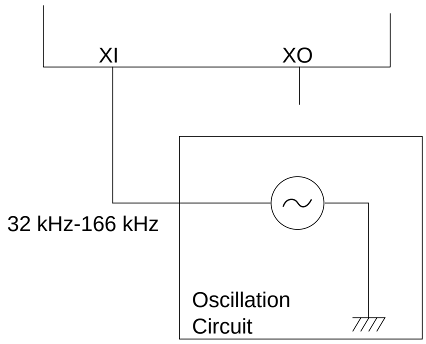




Note : The capacitance values vary depending on the crystal oscillator.

Figure 1-4-10 OSCI, OSCO Connection Example





Note : The capacitance values vary depending on the crystal oscillator.

Figure 1-4-11 XI, XO Connection Example

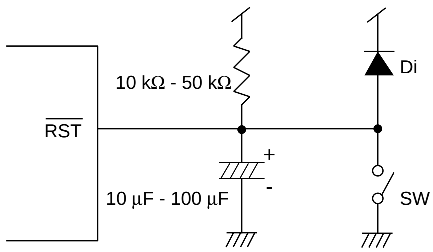


Figure 1-4-12 Reset Pin Connection Example

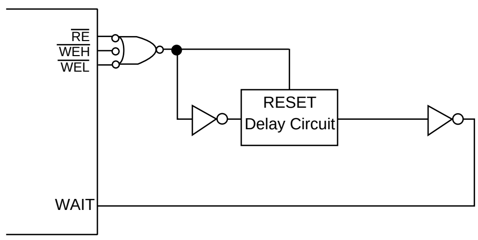
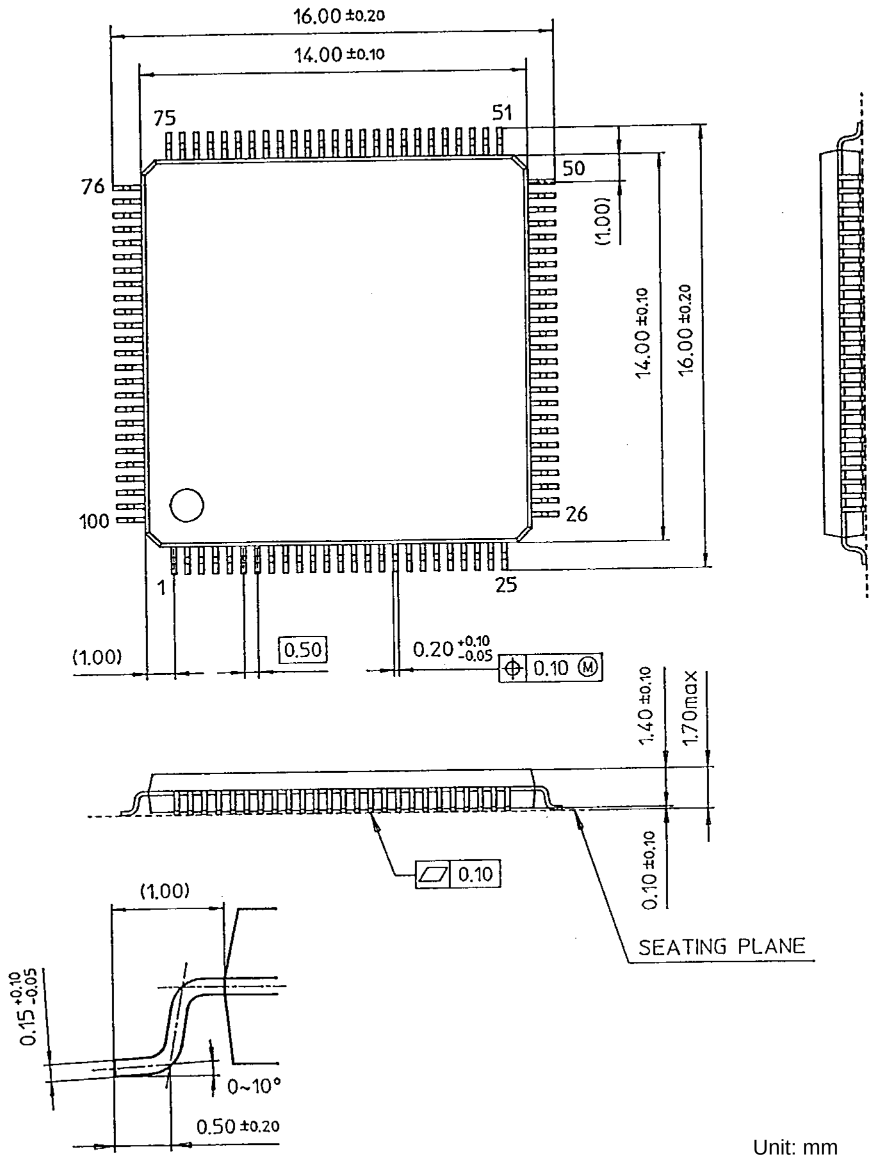


Figure 1-4-13 WAIT Signal Control Circuit Connection Example

Package Code: LQFP100-P-1414



Body Material: Epoxy Resin, Lead Material: FeNi42 Alloy, Lead Finish Method: Solder Plating

Figure 1-4-14 External Dimensions : 100-pin LQFP

External dimensions are subject to change. Before using, please contact your nearest sales office for the latest product specifications.

Chapter 2 Bus Interface

2

2-1 Summary of Bus Interface

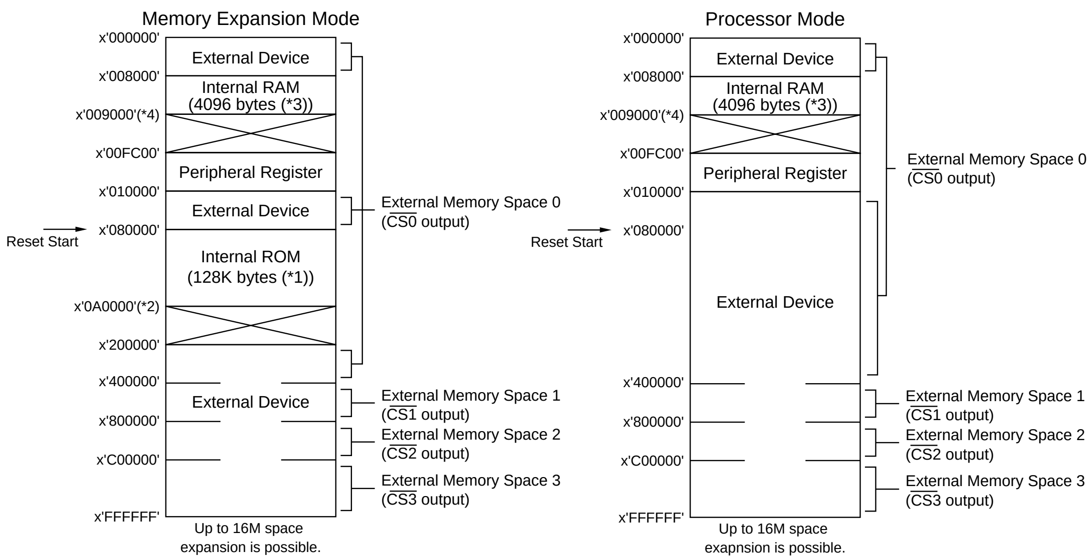
2-1-1 Overview

The MN102H55D/55G/F55G has a function to expand memory to external devices. Table 2-1-1 shows memory modes. 8 or 16-bit data bus width can be selected by setting pins.

Table 2-1-1 Mode Setting

Modes	External Connecting Modes	External Data Bus Width	MODE	ADSEP	WORD	P0MD-P6MD Registers
Single-chip mode	-	-	H	-	-	-
Memory Expansion	Address/data separate mode	8-bit	H	H	H	Note 1
		16-bit	H	H	L	Note 1
	Address/data shared mode	8-bit	H	L	H	Note 1
		16-bit	H	L	L	Note 1
Processor mode	Address/data separate mode	8-bit	L	H	H	Note 2
		16-bit	L	H	L	Note 2
	Address/data shared mode	8-bit	L	L	H	Note 2
		16-bit	L	L	L	Note 2

Note 1: Set each mode register to input or output an address/data control signal from single-chip mode using user program on internal ROM becuase the CPU starts in single-chip mode after reset.
Note 2: Initialize the setting to input or output an address/data control signal after reset release.



	*1	*2	*3	*4
MN102HF55G	128K bytes	x'0A0000'	4096 bytes	x'009000'
MN102H55G	128Kbytes	x'0A0000'	4096 bytes	x'009000'
MN102H55D	64Kbytes	x'090000'	4096 bytes	x'009000'

Figure 2-1-1 Address Space

2-1-2 Control Registers

These registers control the bus interface: the external memory wait register (EXWMD), the memory mode setup 1 register (MEMMD1), the memory mode setup 2 register (MEMMD2), the DRAM control 1 register (DRAMMD1), the DRAM control 2 register (DRAMMD2), the RE waveform control register (REEDGE), the WE waveform control register (WEEDGE), the ALE waveform control register (ALEEDGE) and the address output time control register (MPXADR).

The EXWMD register sets the number of waits for devices in the external memory spaces 0 to 3.

EXWMD: x'00FF80'

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EW33	EW32	EW31	EW30	EW23	EW22	EW21	EW20	EW13	EW12	EW11	EW10	EW03	EW02	EW01	EW00
Reset:	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0

EW[03:00]	Set the number of wait for external memory space 0*
EW[13:10]	Set the number of wait for external memory space 1*
EW[23:20]	Set the number of wait for external memory space 2*
EW[33:30]	Set the number of wait for external memory space 3*

* Please refer to Figure 2-1-1 Address Space on page 52 for address allocation of the external memory spaces.

	Wait	Cycle
0000	0.0	1.0
0001	0.5	1.5
0010	1.0	2.0
0011	1.5	2.5
0100	2.0	3.0
0101	2.5	3.5
0110	3.0	4.0
0111	3.5	4.5
1000	4.0	5.0
1001	4.5	5.5
1010	5.0	6.0
1011	5.5	6.5
1100	6.0	7.0
1101	6.5	7.5
1110	7.0	8.0
1111	perform handshake mode by WAIT pin	

0.5 wait cycle corresponds to BOSC 1 cycle. 1 wait corresponds to 1 cycle of instruction. With a 34-MHz external oscillator,

0.5 wait: 29.4 ns
1.0 wait: 58.8 ns

The MEMMD1 register sets the wait cycles for internal peripherals, the bus widths and ROM burst modes for the external memory spaces 0 to 3.

MEMMD1: x'00FF82'

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EB31	EB30	EB21	EB20	EB11	EB10	EB01	EB00	BRS1	BRS0	BRC3	BRC2	BRC1	BRC0	IOW1	IOW0
Reset	0	0	0	0	0	0	–	–	0	0	0	0	0	0	1	1

IOW[1:0]	Wait Setting for Internal I/O Space
00	1 wait cycle
01	1.5 wait cycles
10	2 wait cycles
11	3 wait cycles

BRC[3:0]	Burst ROM Setting for External Memory Space 3-0
0	Disable
1	Enable

BRS[1:0]	Address Setting for Burst Operation	16-bit Bus Width	8-bit Bus Width
00	A0, A1	2 words	4 bytes
01	A0, A1, A2	4 words	8 bytes
10	A0, A1, A2, A3	8 words	16 bytes
11	A0, A1, A2, A3, A4	16 words	32 bytes

EB[01:00]	Bus Width Setting for External Memory Space 0 (*1,*2)
00	16-bit bus width
01	8-bit bus width
10	Reserved
11	8-bit bus width when A8 pin is high 16-bit bus width when A8 pin is low

*2 The reset value is '00' when /WORD pin is low while the reset value is '01' when /WORD pin is high.

EB[11:10]	Bus Width Setting for External Memory Space 1 (*1)
00	16-bit bus width
01	8-bit bus width
10	Reserved
11	8-bit bus width when A8 pin is high 16-bit bus width when A8 pin is low

EB[21:20]	Bus Width Setting for External Memory Space 2 (*1)
00	16-bit bus width
01	8-bit bus width
10	Reserved
11	8-bit bus width when A8 pin is high 16-bit bus width when A8 pin is low

EB[31:30]	Bus Width Setting for External Memory Space 3 (*1)
00	16-bit bus width
01	8-bit bus width
10	Reserved
11	8-bit bus width when A8 pin is high 16-bit bus width when A8 pin is low

*1 Please refer to Figure 2-1-1 Memory Space on page 52 for address allocation of the external memory spaces.

Do not access the burst ROM area and other areas consecutively.

The MEMMD2 register sets the cycles during burst ROM mode and the fixed wait cycles during handshake mode.

MEMMD2: x'00FF84'

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	HS2	HS1	HS0	—	—	reserved	reserved	—	BST2	BST1	BST0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

BST[2:0]	Cycle Setting for Burst ROM Shortening (First Cycle Setting at Burst Access)
000	0.5 cycle
001	1.0 cycle
010	1.5 cycles
011	2.0 cycles
100	2.5 cycles
101	3.0 cycles
110	3.5 cycles
111	4.0 cycles

Reserved	Set to 0

HS[2:0]	Fixed Wait Setting When Controlling Wait Cycles in WAIT Pin Handshake Mode
000	No wait
001	0.5 wait cycle
010	1 wait cycle
011	1.5 wait cycles
100	2 wait cycles
101	2.5 wait cycles
110	3 wait cycles
111	3.5 wait cycles

The DRAMMD1 register sets the external memory DRAM operation, the timing of $\overline{RAS}$ and $\overline{CAS}$, and the size of address shift.

DRAMMD1: x'00FF90'

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ARE3	ARE2	ARE1	ARE0	MMD1	MMD0	ASFN	SEL2	SEL1	SEL0	CAS2	CAS1	CAS0	RAS2	RAS1	RAS0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

RAS[2:0]	Timing Setting of the $\overline{RAS}$'s Falling Edge
000	At the beginning of 0.5 cycle
001	At the beginning of 1.0 cycle
010	At the beginning of 1.5 cycles
011	At the beginning of 2.0 cycles
100	At the beginning of 2.5 cycles
101	At the beginning of 3.0 cycles
110	At the beginning of 3.5 cycles
111	At the beginning of 4.0 cycles

CAS[2:0]	Timing Setting of the $\overline{CAS}$'s Falling Edge
000	At the beginning of 0.5 cycle
001	At the beginning of 1.0 cycle
010	At the beginning of 1.5 cycles
011	At the beginning of 2.0 cycles
100	At the beginning of 2.5 cycles
101	At the beginning of 3.0 cycles
110	At the beginning of 3.5 cycles
111	At the beginning of 4.0 cycles

SEL[2:0]	Timing Setting of Shifting from Row address to Column Address
000	At the beginning of 0.5 cycle
001	At the beginning of 1.0 cycle
010	At the beginning of 1.5 cycles
011	At the beginning of 2.0 cycles
100	At the beginning of 2.5 cycles
101	At the beginning of 3.0 cycles
110	At the beginning of 3.5 cycles
111	At the beginning of 4.0 cycles

ASFN	Shift Setting from Row Address of AD15 - AD0 pins to Column Address
0	Do not shift
1	Shift

MMD[1:0]	Shift Size of DRAM Address
00	8-bit
01	9-bit
10	10-bit
11	Reserved

MMD(1:0) Setting		00 Shift 8	01 Shift 9	10 Shift 10
Pin Name	ROW Address Output	COL Address Output		
P46	A22	-	(A11)	(A11)
P45	A21	-	(A10)	(A10)
P44	A20	-	(Lo)	A10
P43	A19	-	(Lo)	A9
P42	A18	-	A9	A8
P41	A17	-	A8	A7
P40	A16	A8	A7	A6
P37	A15	A7	A6	A5
P36	A14	A6	A5	A4
P35	A13	A5	A4	A3
P34	A12	A4	A3	A2
P33	A11	A3	A2	A1
P32	A10	A2	A1	A0
P31	A9	A1	A0	(A0)
P30	A8	A0	-	-

ARE[3:0]	DRAM Operation for External Memory Space 3-0*
0	Disable
1	Enable

* Please refer to Figure 2-1-1 Memory Space on page 52 for address allocation of the external memory spaces.

The DRAMMD2 register sets the DRAM refresh operation, the refresh timing and the access method.

DRAMMD2: x'00FF92'

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRAM ACC	DRAM TM	reserv ed	reserv ed	reserv ed	RON	RCY3	RCY2	RCY1	RCY0	RCS2	RCS1	RCS0	RRS2	RRS1	RRS0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

RRS[2:0]	Timing Setting of the $\overline{RAS}$'s Falling Edge at Refresh
000	At the beginning of 0.5 cycle
001	At the beginning of 1.0 cycle
010	At the beginning of 1.5 cycles
011	At the beginning of 2.0 cycles
100	At the beginning of 2.5 cycles
101	At the beginning of 3.0 cycles
110	At the beginning of 3.5 cycles
111	At the beginning of 4.0 cycles

RCS[2:0]	Timing Setting of the $\overline{CAS}$'s Falling Edge at Refresh
000	At the beginning of 0.5 cycle
001	At the beginning of 1.0 cycle
010	At the beginning of 1.5 cycles
011	At the beginning of 2.0 cycles
100	At the beginning of 2.5 cycles
101	At the beginning of 3.0 cycles
110	At the beginning of 3.5 cycles
111	At the beginning of 4.0 cycles

RCY[3:0]	Cycle Setting at Refresh
0000	2.0 cycles
0001	2.5 cycles
0010	3.0 cycles
0011	3.5 cycles
0100	4.0 cycles
0101	4.5 cycles
0110	5.0 cycles
0111	5.5 cycles
1000	6.0 cycles
1001	6.5 cycles
Others	7.0 cycles

RON	DRAM Refresh Enable *
0	Disable
1	Enable

* DRAM is refreshed once when the timer 10/12 underflow interrupt is generated.
At 256 times/8 ms, the refresh interval is 31.25 µs or less.

Reserved	Set to 0.

DRAMTM	Clock Source Selection for DRAM Refresh
0	Timer 12 underflow
1	Timer 10 underflow

DRAMACC	DRAM Access Method Selection
0	2WE method
1	Reserved

The REEDGE register sets the /RE waveform control modes for the external memory spaces 0 to 3.

REEDGE: x'00FF86'

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	REES 31	REES 30	REEL 31	REEL 30	REES 21	REES 20	REEL 21	REEL 20	REES 11	REES 10	REEL 11	REEL 10	REES 01	REES 00	REEL 01	REEL 00
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

REEL[01:00]	$\overline{\text{RE}}$ Late Mode in $\overline{\text{CS0}}$ Area
00	$\overline{\text{RE}}$ late 0.5 mode (at reset)
01	$\overline{\text{RE}}$ late 1 mode
10	$\overline{\text{RE}}$ late 2 mode
11	$\overline{\text{RE}}$ late 3 mode

REES[01:00]	$\overline{\text{RE}}$ Short Mode in $\overline{\text{CS0}}$ Area
00	$\overline{\text{RE}}$ short 0 mode
01	$\overline{\text{RE}}$ short 0.5 mode (at reset)
10	$\overline{\text{RE}}$ short 1 mode
11	$\overline{\text{RE}}$ short 1.5 mode

REEL[11:10]	$\overline{\text{RE}}$ Late Mode in $\overline{\text{CS1}}$ Area
00	$\overline{\text{RE}}$ late 0.5 mode (at reset)
01	$\overline{\text{RE}}$ late 1 mode
10	$\overline{\text{RE}}$ late 2 mode
11	$\overline{\text{RE}}$ late 3 mode

REES[11:10]	$\overline{\text{RE}}$ Short Mode in $\overline{\text{CS1}}$ Area
00	$\overline{\text{RE}}$ short 0 mode (at reset)
01	$\overline{\text{RE}}$ short 0.5 mode
10	$\overline{\text{RE}}$ short 1 mode
11	$\overline{\text{RE}}$ short 1.5 mode

REEL[21:20]	$\overline{\text{RE}}$ Late Mode in $\overline{\text{CS2}}$ Area
00	$\overline{\text{RE}}$ late 0.5 mode (at reset)
01	$\overline{\text{RE}}$ late 1 mode
10	$\overline{\text{RE}}$ late 2 mode
11	$\overline{\text{RE}}$ late 3 mode

REES[21:20]	$\overline{\text{RE}}$ Short Mode in $\overline{\text{CS2}}$ Area
00	$\overline{\text{RE}}$ short 0 mode (at reset)
01	$\overline{\text{RE}}$ short 0.5 mode
10	$\overline{\text{RE}}$ short 1 mode
11	$\overline{\text{RE}}$ short 1.5 mode

REEL[31:30]	$\overline{\text{RE}}$ Late Mode in $\overline{\text{CS3}}$ Area
00	$\overline{\text{RE}}$ late 0.5 mode (at reset)
01	$\overline{\text{RE}}$ late 1 mode
10	$\overline{\text{RE}}$ late 2 mode
11	$\overline{\text{RE}}$ late 3 mode

REES[31:30]	$\overline{\text{RE}}$ Short Mode in $\overline{\text{CS3}}$ Area
00	$\overline{\text{RE}}$ short 0 mode (at reset)
01	$\overline{\text{RE}}$ short 0.5 mode
10	$\overline{\text{RE}}$ short 1 mode
11	$\overline{\text{RE}}$ short 1.5 mode

Please refer to Table 2-2-2 on page 76 and Table 2-2-4 on page 78 for the timing.

The WEEDGE register sets the WE waveform control modes for the external memory spaces 0 to 3.

WEEDGE: x'00FF88'

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WEES31	WEES30	WEEL31	WEEL30	WEES21	WEES20	WEEL21	WEEL20	WEES11	WEES10	WEEL11	WEEL10	WEES01	WEES00	WEEL01	WEEL00
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

WEEL[01:00]	$\overline{\text{WE}}$ Late Mode in $\overline{\text{CS0}}$ Area
00	$\overline{\text{WE}}$ late 1 mode (at reset)
01	$\overline{\text{WE}}$ late 2 mode
10	$\overline{\text{WE}}$ late 3 mode
11	Reserved

WEES[01:00]	$\overline{\text{WE}}$ Short Mode in $\overline{\text{CS0}}$ Area
00	WE short 0 mode (at reset)
01	WE short 0.5 mode
10	WE short 1 mode
11	WE short 1.5 mode

WEEL[11:10]	$\overline{\text{WE}}$ Late Mode in $\overline{\text{CS1}}$ Area
00	$\overline{\text{WE}}$ late 1 mode (at reset)
01	$\overline{\text{WE}}$ late 2 mode
10	$\overline{\text{WE}}$ late 3 mode
11	Reserved

WEES[11:10]	$\overline{\text{WE}}$ Short Mode in $\overline{\text{CS1}}$ Area
00	$\overline{\text{WE}}$ short 0 mode (at reset)
01	$\overline{\text{WE}}$ short 0.5 mode
10	$\overline{\text{WE}}$ short 1 mode
11	$\overline{\text{WE}}$ short 1.5 mode

WEEL[21:20]	$\overline{\text{WE}}$ Late Mode in $\overline{\text{CS2}}$ Area
00	$\overline{\text{WE}}$ late 1 mode (at reset)
01	$\overline{\text{WE}}$ late 2 mode
10	$\overline{\text{WE}}$ late 3 mode
11	Reserved

WEES[21:20]	$\overline{\text{WE}}$ Short Mode in $\overline{\text{CS2}}$ Area
00	$\overline{\text{WE}}$ short 0 mode (at reset)
01	$\overline{\text{WE}}$ short 0.5 mode
10	$\overline{\text{WE}}$ short 1 mode
11	$\overline{\text{WE}}$ short 1.5 mode

WEEL[31:30]	$\overline{\text{WE}}$ Late Mode in $\overline{\text{CS3}}$ Area
00	$\overline{\text{WE}}$ late 1 mode (at reset)
01	$\overline{\text{WE}}$ late 2 mode
10	$\overline{\text{WE}}$ late 3 mode
11	Reserved

WEES[31:30]	$\overline{\text{WE}}$ Short Mode in $\overline{\text{CS3}}$ Area
00	$\overline{\text{WE}}$ short 0 mode (at reset)
01	$\overline{\text{WE}}$ short 0.5 mode
10	$\overline{\text{WE}}$ short 1 mode
11	$\overline{\text{WE}}$ short 1.5 mode

Please refer to Table 2-2-3 on page 77 and Table 2-2-5 on page 78 for the timing.

The ALEEDGE register sets the /RE waveform control modes for the external memory spaces 0 to 3 during address/data shared mode.

ALEEDGE: x'00FF8A'

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALEG 31	ALEG 30	ALEL 31	ALEL 30	ALEG 21	ALEG 20	ALEL 21	ALEL 20	ALEG 11	ALEG 10	ALEL 11	ALEL 10	ALEG 01	ALEG 00	ALEL 01	ALEL 00
Reset	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0

ALEL[01:00]	ALE Late Mode in $\overline{CS0}$ Area
00	ALE late 0 mode
01	ALE late 0.5 mode
10	ALE late 1 mode (at reset)
11	ALE late 1.5 mode

ALEG[01:00]	ALE Long Mode in $\overline{CS0}$ Area
00	ALE long 0 mode
01	ALE long 0.5 mode
10	ALE long 1 mode (at reset)
11	ALE long 1.5 mode

ALEL[11:10]	ALE Late Mode in $\overline{CS1}$ Area
00	ALE late 0 mode (at reset)
01	ALE late 0.5 mode
10	ALE late 1 mode
11	ALE late 1.5 mode

ALEG[11:10]	ALE Long Mode in $\overline{CS1}$ Area
00	ALE long 0 mode (at reset)
01	ALE long 0.5 mode
10	ALE long 1 mode
11	ALE long 1.5 mode

ALEL[21:20]	ALE Late Mode in $\overline{CS2}$ Area
00	ALE late 0 mode (at reset)
01	ALE late 0.5 mode
10	ALE late 1 mode
11	ALE late 1.5 mode

ALEG[21:20]	ALE Long Mode in $\overline{CS2}$ Area
00	ALE long 0 mode (at reset)
01	ALE long 0.5 mode
10	ALE long 1 mode
11	ALE long 1.5 mode

ALEL[31:30]	ALE Late Mode in $\overline{CS3}$ Area
00	ALE late 0 mode (at reset)
01	ALE late 0.5 mode
10	ALE late 1 mode
11	ALE late 1.5 mode

ALEG[31:30]	ALE Long Mode in $\overline{CS3}$ Area
00	ALE long 0 mode (at reset)
01	ALE long 0.5 mode
10	ALE long 1 mode
11	ALE long 1.5 mode

Please refer to Table 2-2-6 on page 79 for the timing.

The MPXADR register sets the address output timing for the external memory spaces 0 to 3 during address/data shared mode.

MPXADR: x'00FF8C'

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ADL 31	ADL 30	ADL 21	ADL 20	ADL 11	ADL 10	ADL 01	ADL 00
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

ADL[01:00]	Address Long Mode in $\overline{CS0}$ Area
00	AD long 1 mode
01	AD long 1.5 mode
10	AD long 2 mode
11	AD long 3 mode (at reset)

ADL[11:10]	Address Long Mode in $\overline{CS1}$ Area
00	AD long 1 mode (at reset)
01	AD long 1.5 mode
10	AD long 2 mode
11	AD long 3 mode

ADL[21:20]	Address Long Mode in $\overline{CS2}$ Area
00	AD long 1 mode (at reset)
01	AD long 1.5 mode
10	AD long 2 mode
11	AD long 3 mode

ADL[31:30]	Address Long Mode in $\overline{CS2}$ Area
00	AD long 1 mode (at reset)
01	AD long 1.5 mode
10	AD long 2 mode
11	AD long 3 mode

Table 2-1-2 List of Bus Interface Control Registers

Register	Address	R/W	Function
EXWMD	x'00FF80'	R/W	External Memory Wait Register
MEMMD1	x'00FF82'	R/W	Memory Mode Setup 1 Register
MEMMD2	x'00FF84'	R/W	Memory Mode Setup 2 Register
DRAMMD1	x'00FF90'	R/W	DRAM Control 1 Register
DRAMMD2	x'00FF92'	R/W	DRAM Control 2 Register
REEDGE	x'00FF86'	R/W	RE Waveform Control Register
WEEDGE	x'00FF88'	R/W	WE Waveform Control Register
ALEEDGE	x'00FF8A'	R/W	ALE Waveform Control Register
MPXADR	x'00FF8C'	R/W	Address Output Time Control Register

Please refer to Table 2-2-7 on page 79 for the timing.

2-1-3 Memory Connection Examples

The MN102H55D/55G/F55G can connect to SRAM, DRAM, mask ROM or burst ROM. This section shows the connection Examples.

■ Example of SRAM (Mask ROM) Connection (16-bit Bus Width, 1 Wait)

When connecting to mask ROM, do not need to connect $\overline{\text{WEH}}$ and $\overline{\text{WEL}}$.

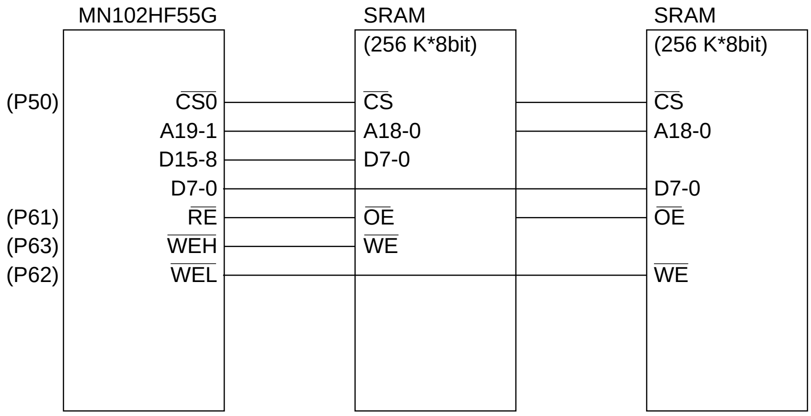


Figure 2-1-2 SRAM (Mask ROM) Connection Example (16-bit Bus Width)

EXWMD	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
													EW 03	EW 02	EW 01	EW 00
													0	0	1	0

MEMMD1	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
							EB 01	EB 00						BRC 0		
							0	0						0		

DRAMMD1	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				ARE 0												
				0												

P0MD	7	6	5	4	3	2	1	0
							P0 MD1	P0 MD0
							0	1

P1LMD	7	6	5	4	3	2	1	0
							P1 LMD1	P1 LMD0
							0	1

P2MD	7	6	5	4	3	2	1	0
								P2 MD0
								1

P3HMD	7	6	5	4	3	2	1	0
	P3 HMD7	P3 HMD6	P3 HMD5	P3 HMD4	P3 HMD3	P3 HMD2	P3 HMD1	P3 HMD0
	0	1	0	1	0	1	0	1

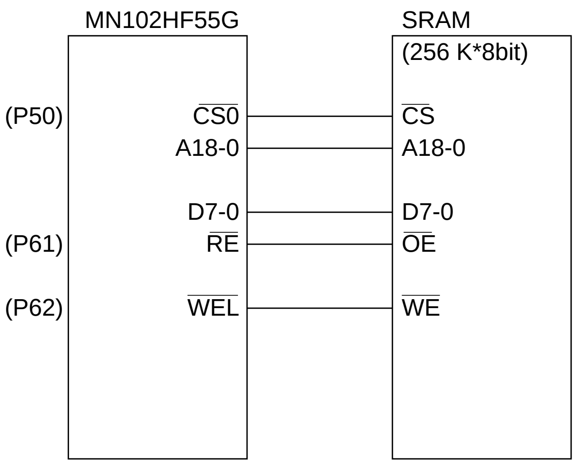
P3LMD	7	6	5	4	3	2	1	0
	P3 LMD7	P3 LMD6	P3 LMD5	P3 LMD4	P3 LMD3	P3 LMD2	P3 LMD1	P3 LMD0
	0	1	0	1	0	1	0	1

P4LMD	7	6	5	4	3	2	1	0
					P4 LMD3	P4 LMD2	P4 LMD1	P4 LMD0
					1	1	1	1

P5LMD	7	6	5	4	3	2	1	0
							P5 LMD1	P5 LMD0
							0	1

P6MD	7	6	5	4	3	2	1	0
			P6 MD5	P6 MD4	P6 MD3			
			1	1	1			

■ Example of SRAM (Mask ROM) Connection (8-bit Bus Width, 1 Wait)



When connecting to mask ROM, do not need to connect $\overline{WEH}$ and $\overline{WEL}$.

Figure 2-1-3 SRAM (Mask ROM) Connection Example (8-bit Bus Width)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXWMD													EW 03	EW 02	EW 01	EW 00
													0	0	1	0

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MEMMD1							EB 01	EB 00						BRC 0		
							0	1						0		

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DRAMMD1				ARE 0												
				0												

	7	6	5	4	3	2	1	0
P0MD							P0 MD1	P0 MD0
							0	1

	7	6	5	4	3	2	1	0
P1LMD							P1 LMD1	P1 LMD0
							0	0

	7	6	5	4	3	2	1	0
P2MD								P2 MD0
								1

	7	6	5	4	3	2	1	0
P3HMD	P3 HMD7	P3 HMD6	P3 HMD5	P3 HMD4	P3 HMD3	P3 HMD2	P3 HMD1	P3 HMD0
	0	1	0	1	0	1	0	1

	7	6	5	4	3	2	1	0
P3LMD	P3 LMD7	P3 LMD6	P3 LMD5	P3 LMD4	P3 LMD3	P3 LMD2	P3 LMD1	P3 LMD0
	0	1	0	1	0	1	0	1

	7	6	5	4	3	2	1	0
P4LMD						P4 LMD2	P4 LMD1	P4 LMD0
						1	1	1

	7	6	5	4	3	2	1	0
P5LMD							P5 LMD1	P5 LMD0
							0	1

	7	6	5	4	3	2	1	0
P6MD			P6 MD5	P6 MD4	P6 MD3			
			0	1	1			

■ Example of DRAM (2WE Method) Connection (16-bit Bus Width, 2 Wait)

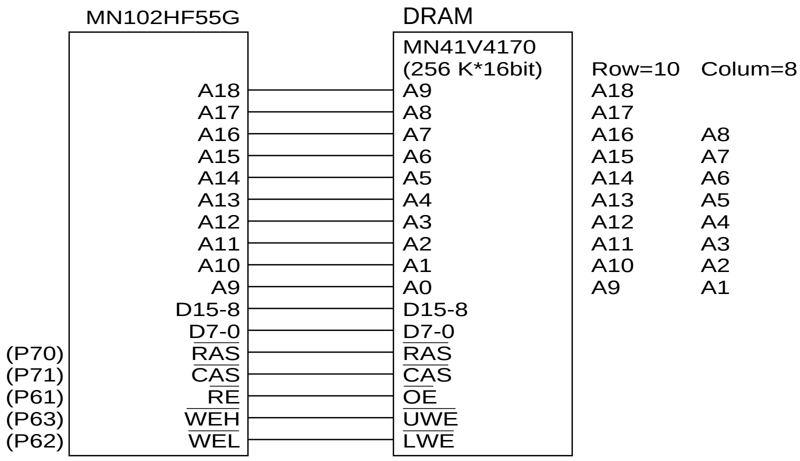


Figure 2-1-4 DRAM (2WE Method) Connection Example (16-bit Bus Width)

EXWMD

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
				EW 23	EW 22	EW 21	EW 20								
				0	1	0	0								

MEMMD1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		EB 21	EB 20								BRC 2				
		0	0								0				

DRAMMD1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ARE 2			MMD 1	MMD 0	AS EN	SEL 2	SEL 1	SEL 0	CAS 2	CAS 1	CAS 0	RAS 2	RAS 1	RAS 0
	1			0	0	1	0	1	0	0	1	1	0	0	1

DRAMMD2

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DRAM ACC	DRAM TM				R ON	RCY 3	RCY 2	RCY 1	RCY 0	RCS 2	RCS 1	RCS 0	RRS 2	RRS 1	RRS 0
0	?	0	0	0	1	0	0	1	0	0	0	0	0	0	1

P0MD

7	6	5	4	3	2	1	0
						P0 MD1	P0 MD0
						0	1

P1LMD

7	6	5	4	3	2	1	0
						P1 LMD1	P1 LMD0
						0	1

P3HMD

7	6	5	4	3	2	1	0
P3 HMD7	P3 HMD6	P3 HMD5	P3 HMD4	P3 HMD3	P3 HMD2	P3 HMD1	P3 HMD0
0	1	0	1	0	1	0	1

P3LMD

7	6	5	4	3	2	1	0
P3 LMD7	P3 LMD6	P3 LMD5	P3 LMD4	P3 LMD3	P3 LMD2		
0	1	0	1	0	1		

P4LMD

7	6	5	4	3	2	1	0
					P4 LMD2	P4 LMD1	P4 LMD0
					1	1	1

P6MD

7	6	5	4	3	2	1	0
		P6 MD5	P6 MD4	P6 MD3			
		1	1	1			

P7LMD

7	6	5	4	3	2	1	0
			P7 LMD4	P7 LMD3	P7 LMD2	P7 LMD1	P7 LMD0
			1	1	1	0	1

■ Example of Burst ROM Connection (8-bit Bus Width, 4-3-3-3 Waits, Lower 2 bits of Address)

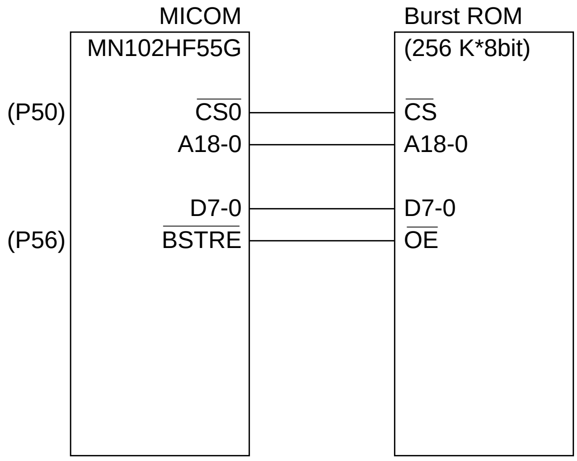


Figure 2-1-5 Burst ROM Connection Example (8-bit Bus Width)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EXWMD													EW 03	EW 02	EW 01	EW 00
													1	0	0	0

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MEMMD1							EB 01	EB 00	BRS 1	BRS 0				BRC 0		
							0	1	0	0				1		

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
MEMMD2														BST 2	BST 1	BST 0
														0	0	1

	7	6	5	4	3	2	1	0
P0MD							P0 MD1	P0 MD0
							0	1

	7	6	5	4	3	2	1	0
P1LMD							P1 LMD1	P1 LMD0
							0	1

	7	6	5	4	3	2	1	0
P2MD								P2 MD0
								1

	7	6	5	4	3	2	1	0
P3HMD	P3 HMD7	P3 HMD6	P3 HMD5	P3 HMD4	P3 HMD3	P3 HMD2	P3 HMD1	P3 HMD0
	0	1	0	1	0	1	0	1

	7	6	5	4	3	2	1	0
P3LMD	P3 LMD7	P3 LMD6	P3 LMD5	P3 LMD4	P3 LMD3	P3 LMD2	P3 LMD1	P3 LMD0
	0	1	0	1	0	1	0	1

	7	6	5	4	3	2	1	0
P4LMD						P4 LMD2	P4 LMD1	P4 LMD0
						1	1	1

	7	6	5	4	3	2	1	0
P5LMD							P5 LMD1	P5 LMD0
							0	1

	7	6	5	4	3	2	1	0
P5HMD				P5 HMD4	P5 HMD3	P5 HMD2		
				0	1	1		

■ Example of DRAM Connection (8-bit Bus Width, 2 Wait)

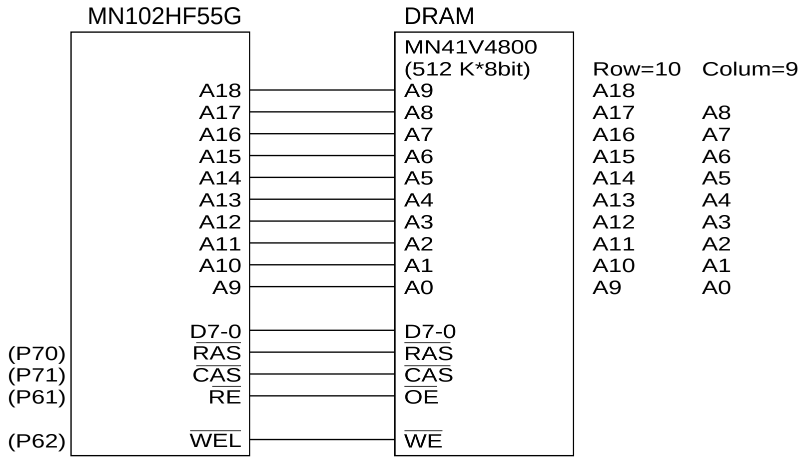


Figure 2-1-6 DRAM Connection Example (8-bit Bus Width)

EXWMD	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
					EW 23	EW 22	EW 21	EW 20								
					0	1	0	0								

MEMMD1	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
			EB 21	EB 20								BRC 2				
			0	1								0				

DRAMMD1	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		ARE 2			MMD 1	MMD 0	AS EN	SEL 2	SEL 1	SEL 0	CAS 2	CAS 1	CAS 0	RAS 2	RAS 1	RAS 0
		1			0	1	1	0	1	0	0	1	1	0	0	1

DRAMMD2	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRAM ACC	DRAM TM				R ON	RCY 3	RCY 2	RCY 1	RCY 0	RCS 2	RCS 1	RCS 0	RRS 2	RRS 1	RRS 0
	1	?	0	0	0	1	1	0	0	1	0	0	0	0	0	1

P0MD	7	6	5	4	3	2	1	0
							P0 MD1	P0 MD0
							0	1

P1LMD	7	6	5	4	3	2	1	0

P2MD	7	6	5	4	3	2	1	0
								P2 MD0
								0

P3HMD	7	6	5	4	3	2	1	0
	P3 HMD7	P3 HMD6	P3 HMD5	P3 HMD4	P3 HMD3	P3 HMD2	P3 HMD1	P3 HMD0
	0	1	0	1	0	1	0	1

P3LMD	7	6	5	4	3	2	1	0
	P3 LMD7	P3 LMD6	P3 LMD5	P3 LMD4	P3 LMD3	P3 LMD2		
	0	1	0	1	0	1		

P4LMD	7	6	5	4	3	2	1	0
						P4 LMD2	P4 LMD1	P4 LMD0
						1	1	1

P6MD	7	6	5	4	3	2	1	0
				P6 MD4	P6 MD3			
				1	1			

P7LMD	7	6	5	4	3	2	1	0
				P7 LMD4	P7 LMD3	P7 LMD2	P7 LMD1	P7 LMD0
				1	1	1	0	1

2-1-4 Access to External Memory

The MN102H55D/55G/F55G can access to external memory. The external memory space is divided into four areas. When the MN102H55D/55G/F55G accesses to each area, the corresponded CSn pin (n=0 to 3) outputs a chip select signal. In addition, the number of wait cycles and 8-bit or 16-bit bus width can be selected for each area.

The clock output from BOSC pin is the base clock at external access. The address, data or control signals output synchronizing with BOSC clock. The BOSC clock frequency is the same as the oscillation clock frequency input from OSCI pin. For example, the BOSC clock frequency become 40 MHz with a 40-MHz external oscillator. (The clock input from OSCI pin and BOSC clock have the phase difference.) The BIBT1 or BIBT2 is the internal clock synchronizing with BOSC clock, and it shows memory access cycle. During 1 memory access, first the BIBT2 clock becomes high level and then the BIBT1 clock becomes high level. When the number of wait cycles is set, the BIBT1 clock cycle being high level will be extended.

When no wait cycle is selected, the BIBT2 clock cycle and the BIBT1 clock cycle being high level equals 1 BOSC cycle. The necessary cycle for 1 access should be 2 BOSC clock cycles.

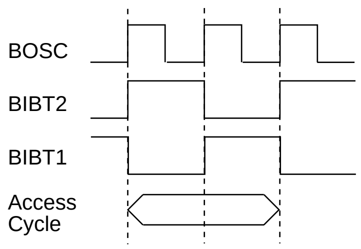


Figure 2-1-7 External Access (No Wait Cycle)

When the number of wait cycles (0.5 wait cycle to 7 wait cycles) is set, the BIBT1 clock cycle being high level is extended 1 BOSC clock cycle in each 0.5 wait cycle. The necessary cycle for 1 access should be 3 BOSC clock cycles.

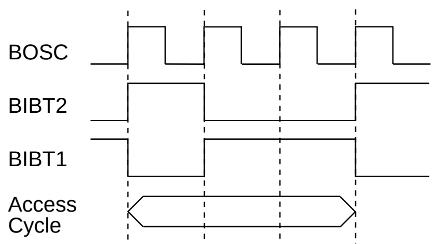


Figure 2-1-8 External Access (0.5 Wait Cycle)

Table 2-1-3 Address/Data Multiplex Mode (16-bit Bus Data Access)

The length of wait cycle can be set in 0.5-cycle units.

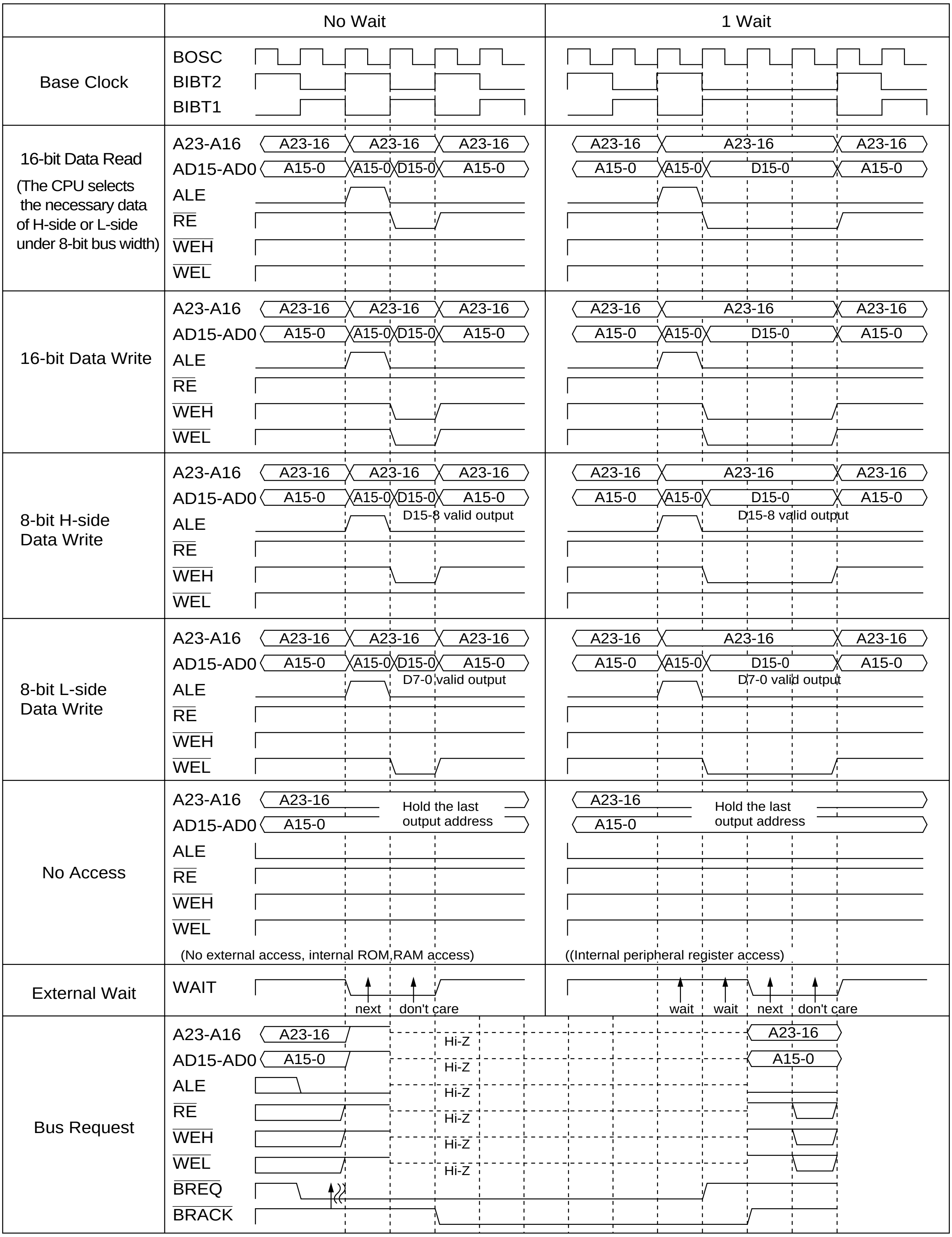


Table 2-1-4 Address/Data Multiplex Mode (8-bit Bus Data Access)

The length of wait cycle can be set in 0.5-cycle units.

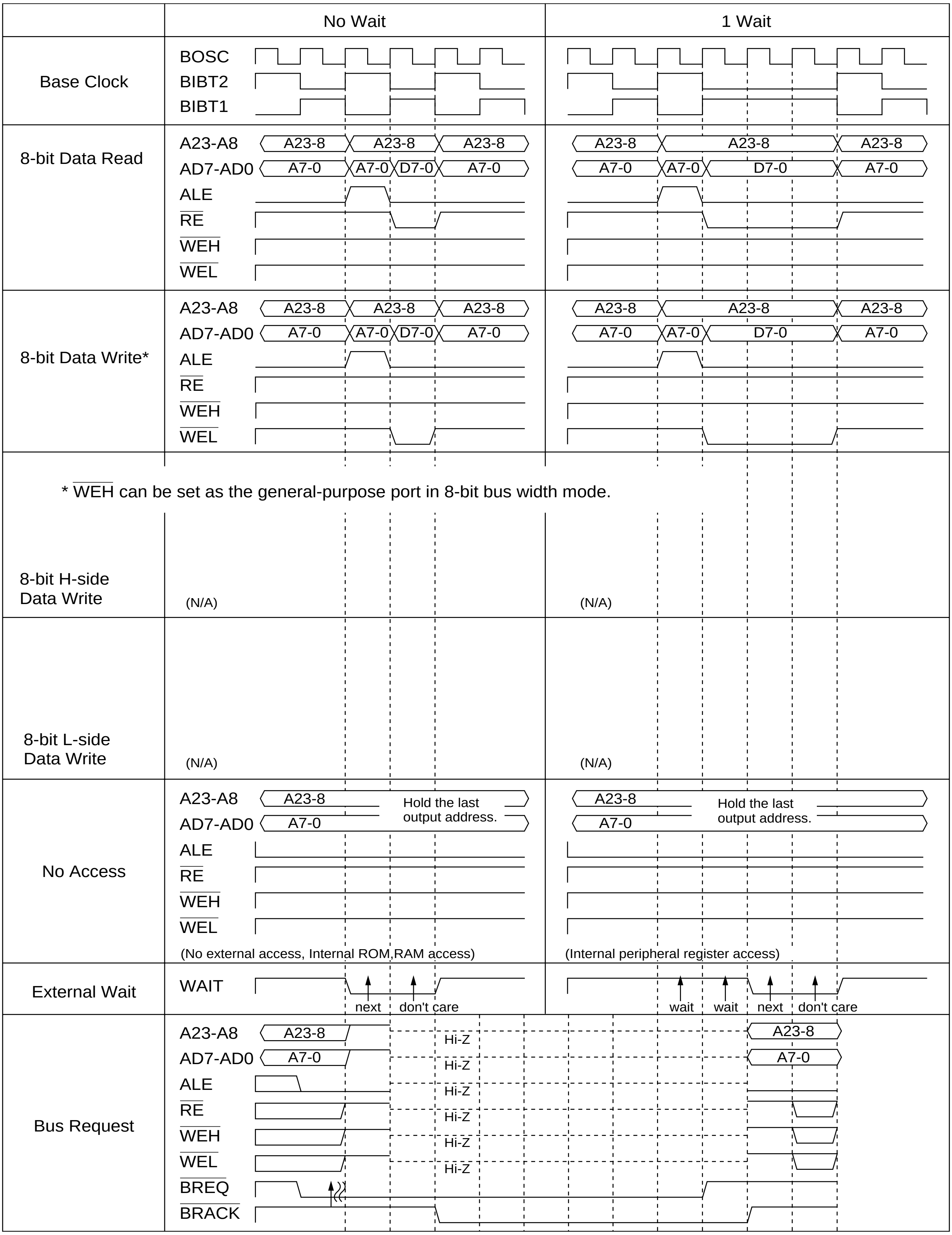


Table 2-1-5 Address/Data Separate Mode (16-bit Bus Data Access)

The length of wait cycle can be set in 0.5-cycle units.

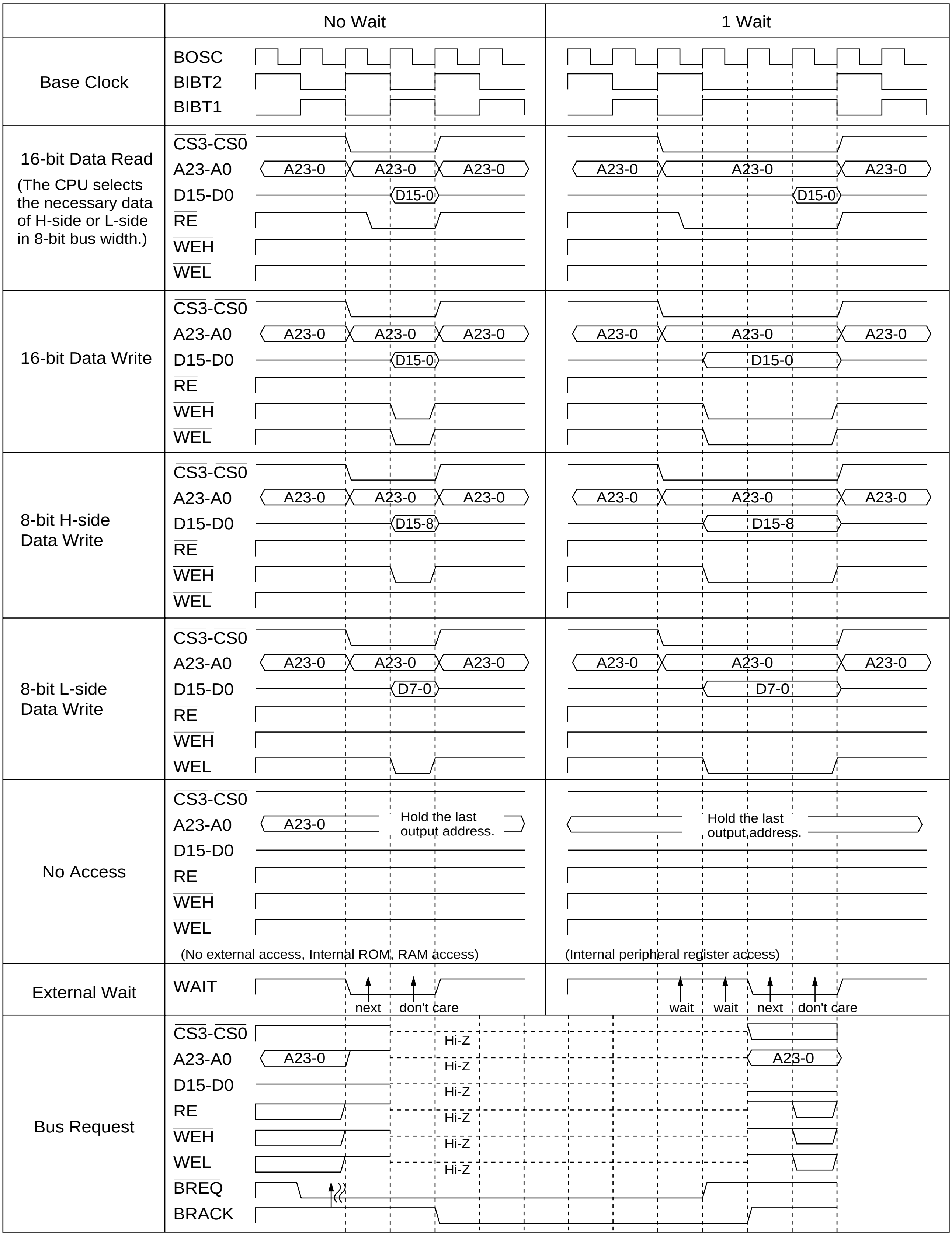


Table 2-1-6 Address/Data Separate Mode (8-bit Bus Data Access)

The length of wait cycle can be set in 0.5-cycle units.

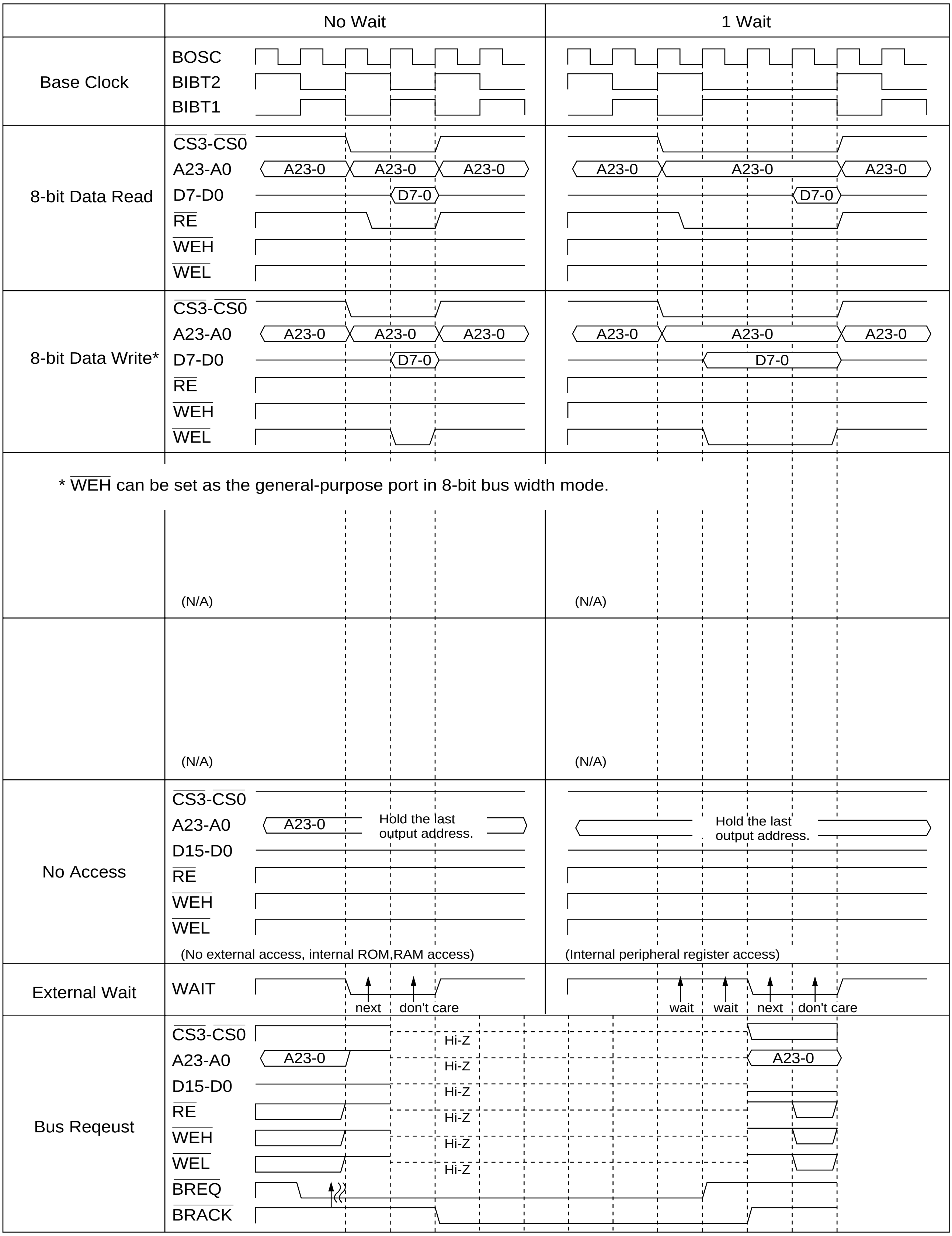
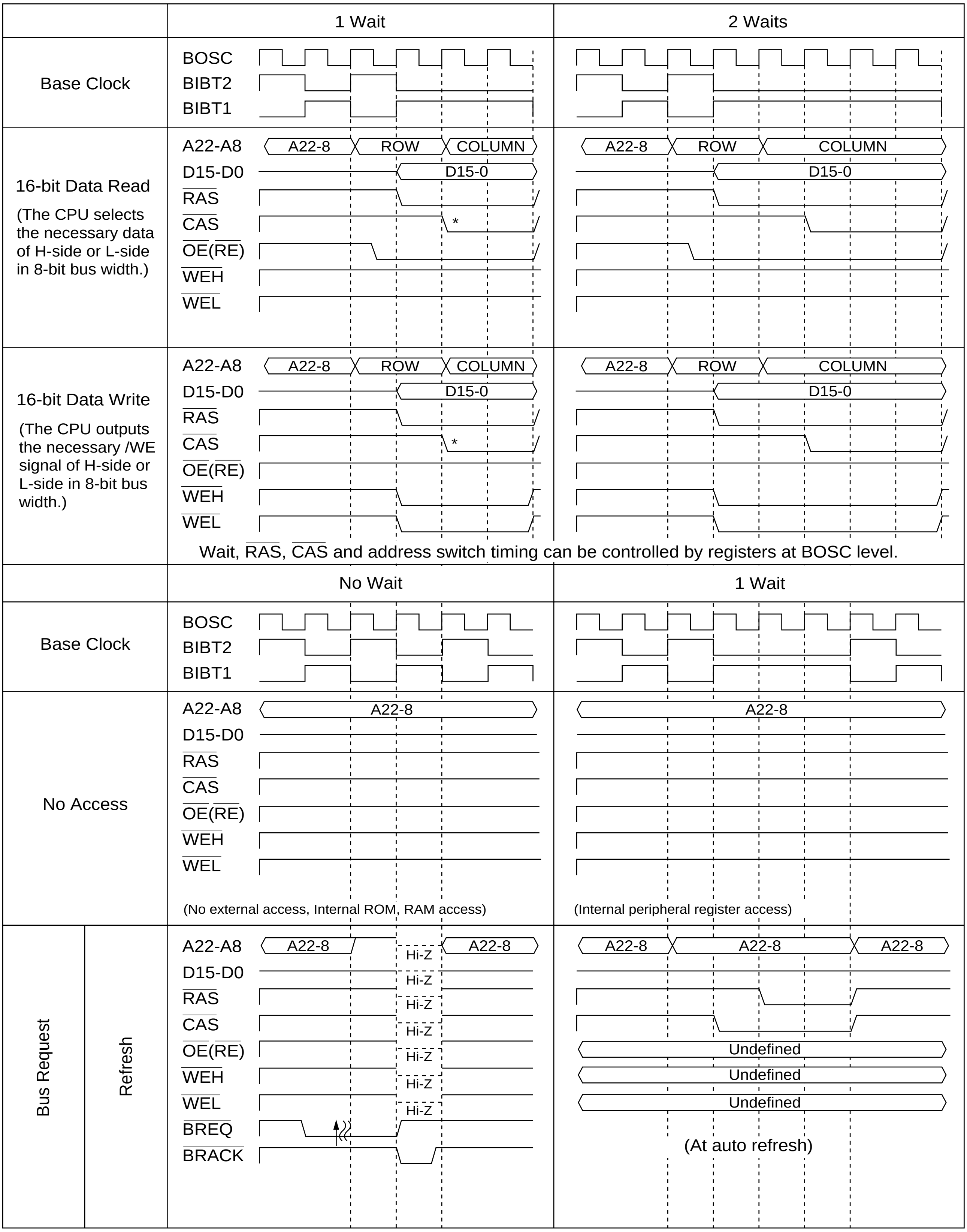


Table 2-1-7 Address/Data Separate Mode (16-bit Bus DRAM, $\overline{\text{WEH}}$ and $\overline{\text{WEL}}$ Method)

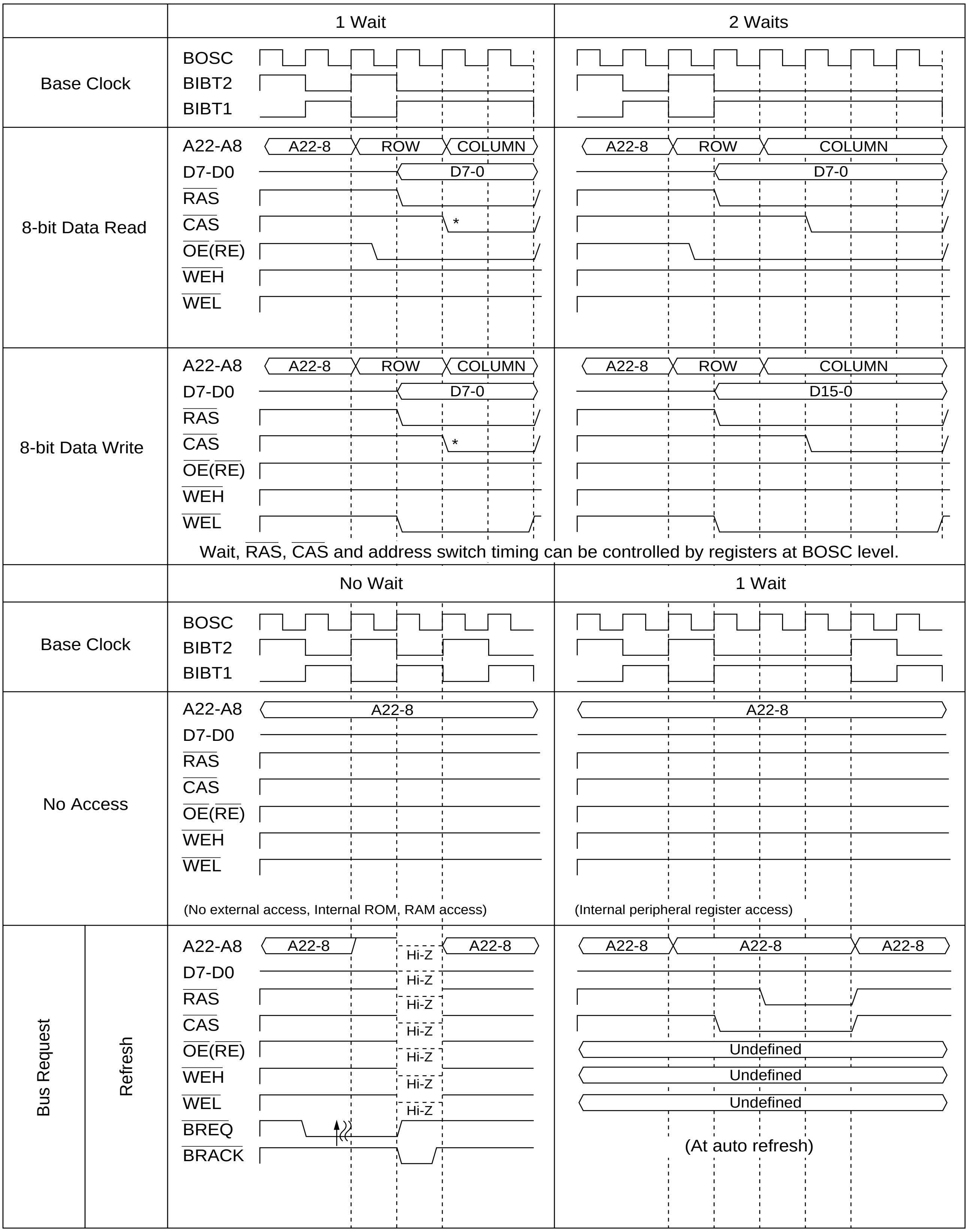
The length of wait cycle can be set in 0.5-cycle units.



* $\overline{\text{CAS}}$ must be delayed externally to hold the setup time of the COLUMN address.

Table 2-1-8 Address/Data Separate Mode (8-bit Bus DRAM, $\overline{\text{WEH}}$ and $\overline{\text{WEL}}$ Method)

The length of wait cycle can be set in 0.5-cycle units.



* $\overline{\text{CAS}}$ must be delayed externally to hold the setup time of the COLUMN address.

Table 2-1-9 Address/Data Separate Mode (16-bit Bus Burst ROM Access)

The length of wait cycle can be set in 0.5-cycle units.

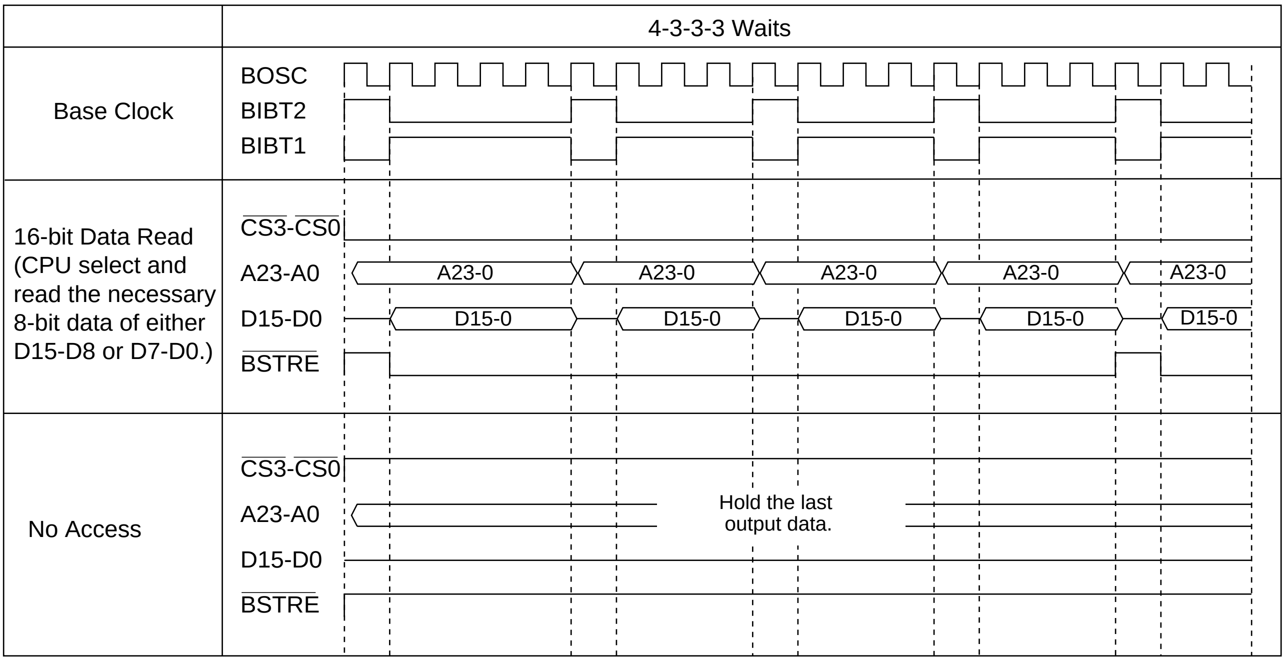
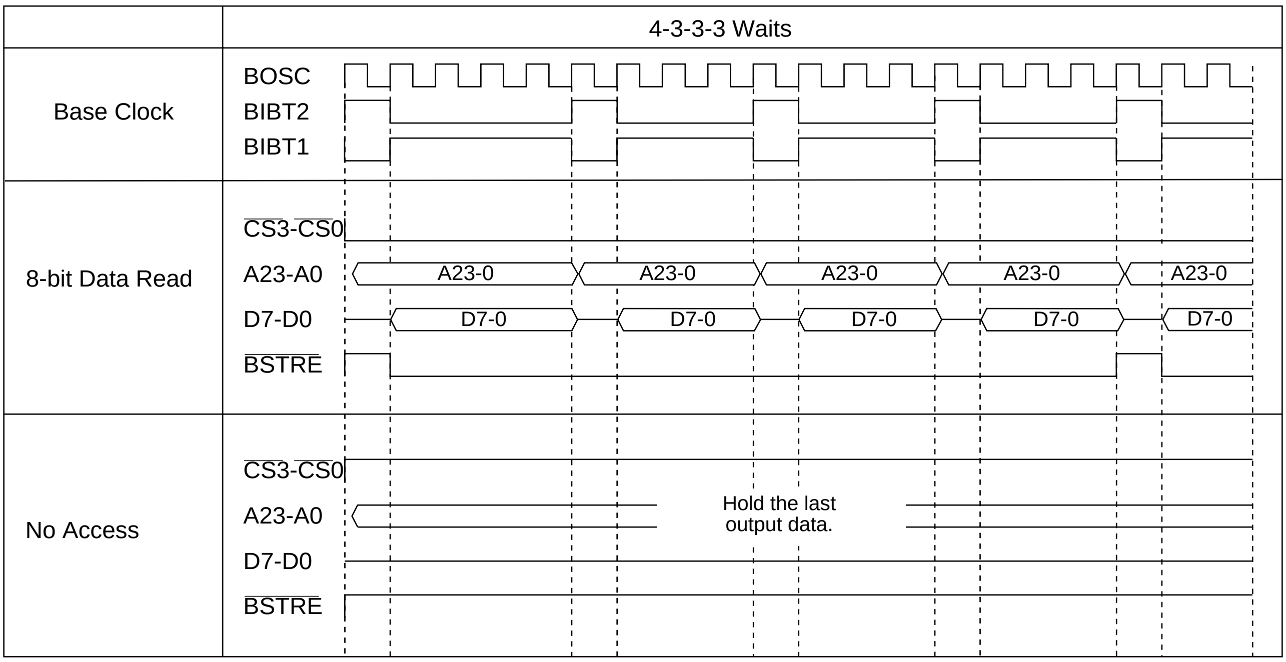


Table 2-1-10 Address/Data Separate Mode (8-bit Bus Burst ROM Access)

The length of wait cycle can be set in 0.5-cycle units.



2-2 Control Signals

2-2-1 Overview

The MN102H55D/55G/F55G can delay or hasten the rising timing and the falling timing of $\overline{\text{RE}}$, $\overline{\text{WE}}$ and ALE waveforms in the external memory extension mode. In addition, it can delay the switching timing of address and data in the address/data shared mode. The $\overline{\text{RE}}$ short mode and the $\overline{\text{RE}}$ late mode do not affect the $\overline{\text{BSTRE}}$ pin connecting burst ROM.

The following table shows settings.

Table 2-2-1 External Memory Control Signal Timing

Signal	Mode	Timing	Function
$\overline{\text{RE}}$	Late	0.5 (Reset) 1, 2, 3	Delay the falling timing of $\overline{\text{RE}}$.
	Short	0, 0.5, 1, 1.5	Hasten the rising timing of $\overline{\text{RE}}$.
$\overline{\text{WE}}$	Late	1 (Reset), 2, 3	Delay the falling timing of $\overline{\text{WE}}$.
	Short	0 (Reset), 0.5, 1, 1.5	Hasten the rising timing of $\overline{\text{WE}}$.
ALE	Late	0, 0.5, 1, 1.5	Delay the rising timing of ALE.
	Short	0, 0.5, 1, 1.5	Hasten the falling timing of ALE.
Address Data	Long	1, 1.5, 2, 3	Delay the switch timing of address and data in the address/data shared mode.

Please refer to page 76 to page 79 for the waveform in each mode.

Table 2-2-2 $\overline{\text{RE}}$ Late and Short Modes (Address/Data Shared Mode)

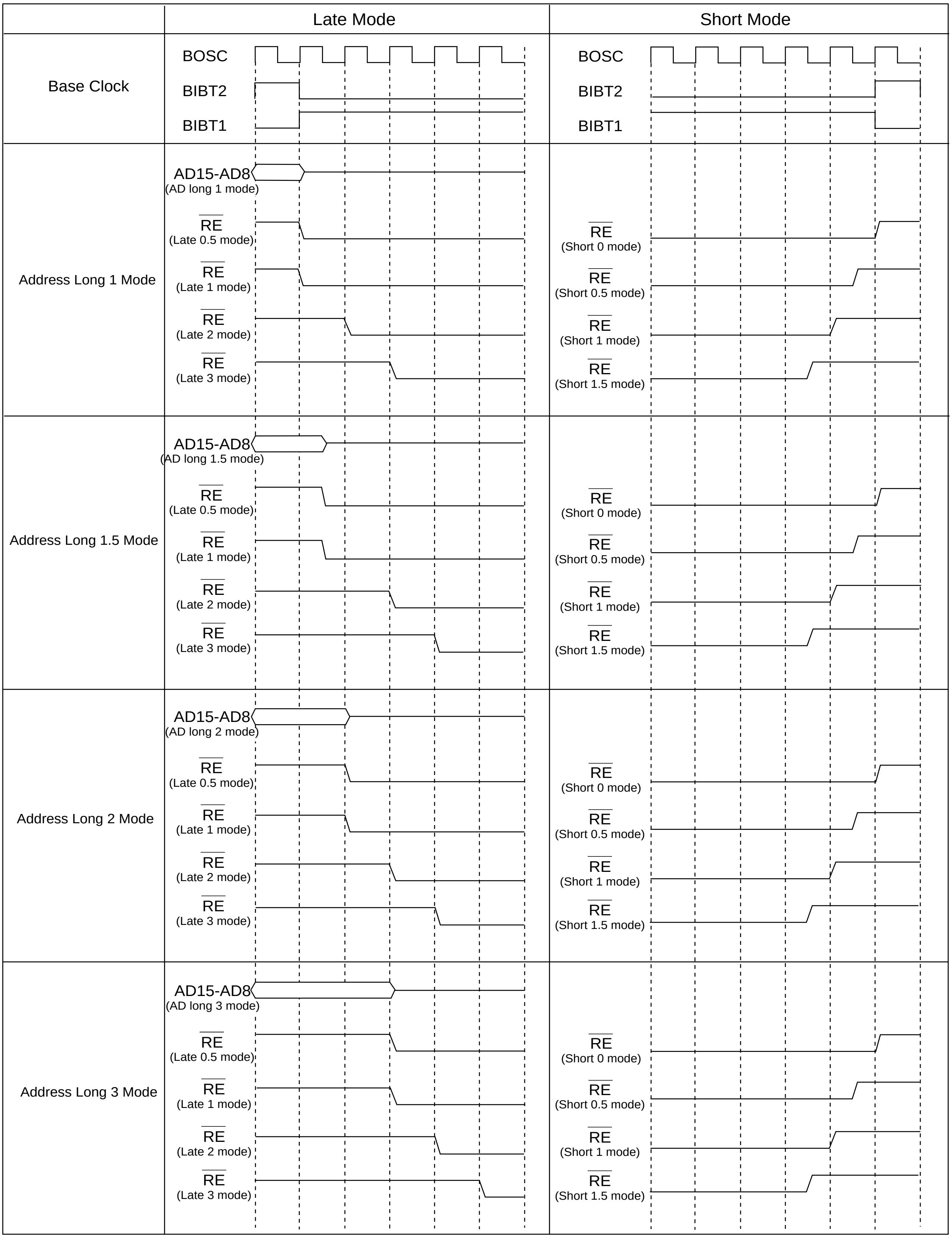


Table 2-2-3 $\overline{WE}$ Late and Short Modes (Address/Data Shared Mode)

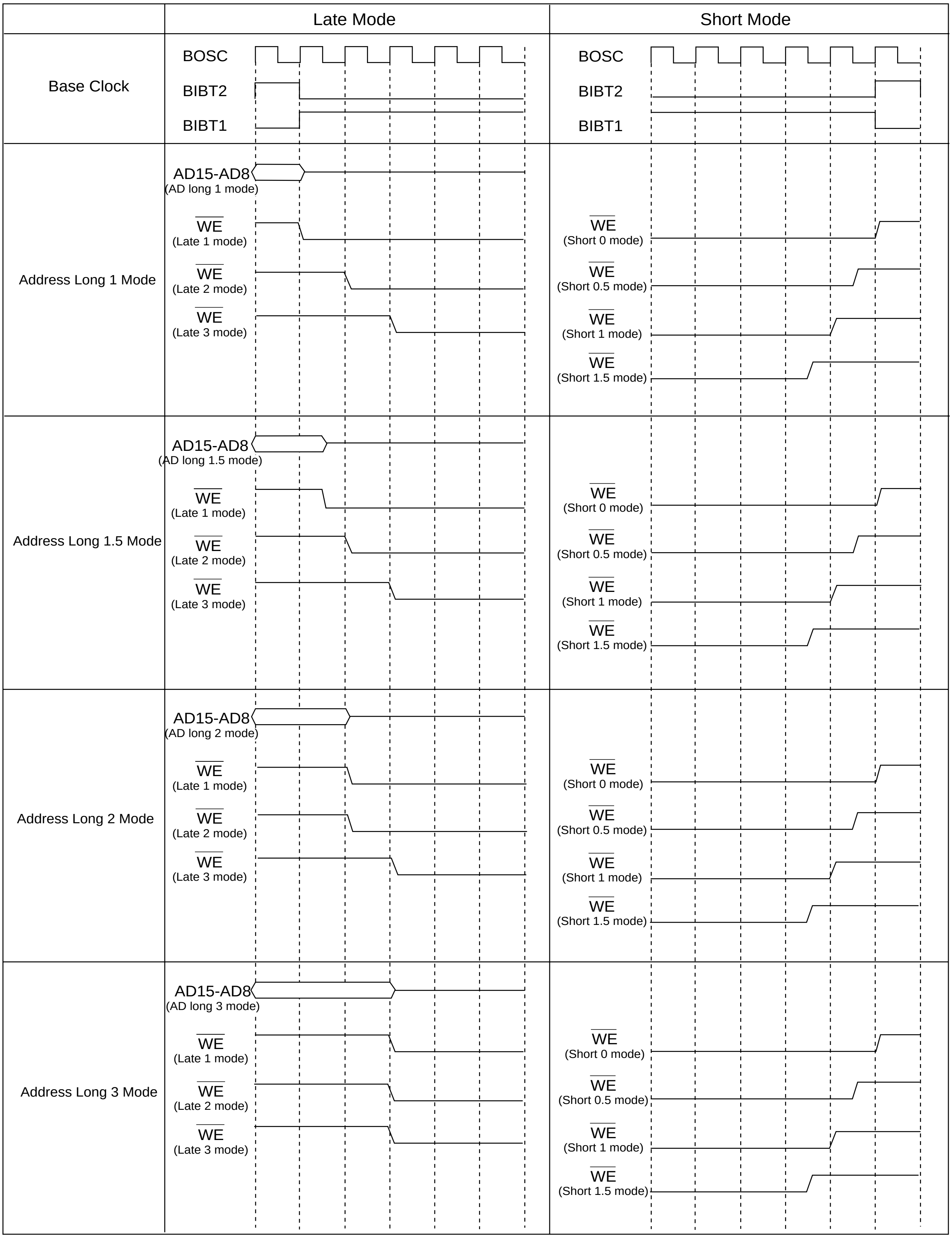


Table 2-2-4 $\overline{RE}$ Late and Short Modes (Address/Data Separate Mode)

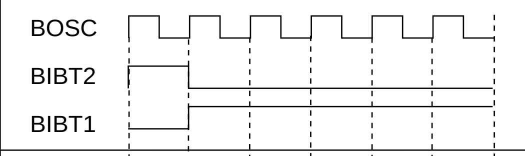
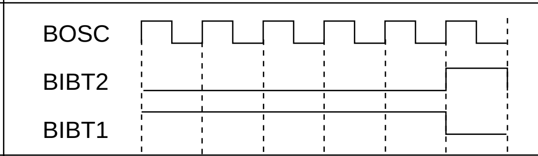
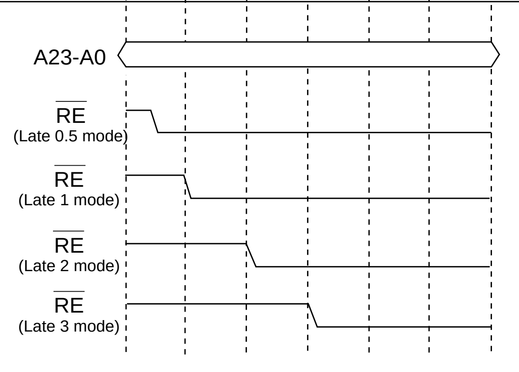
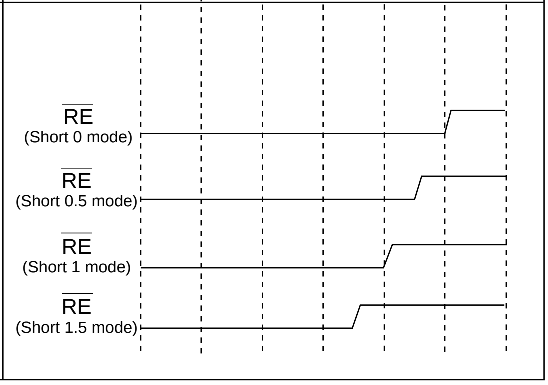
	Late Mode	Short Mode
Base Clock		
		

Table 2-2-5 /WE Late and Short Modes (Address/Data Separate Mode)

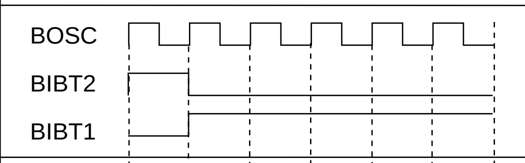
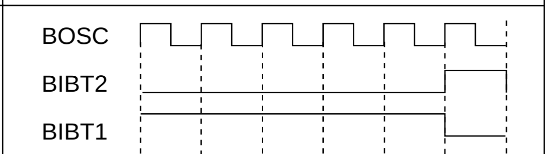
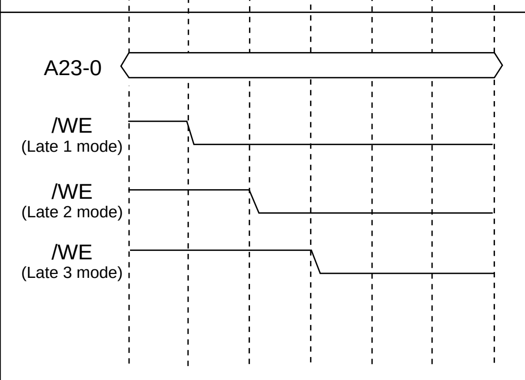
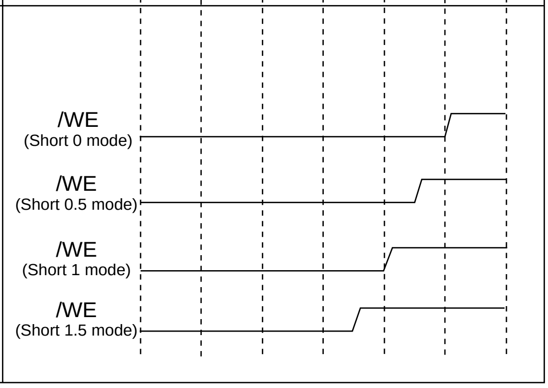
	Late Mode	Short Mode
Base Clock		
		

Table 2-2-6 ALE Late and Long Modes (Address/Data Shared Mode)

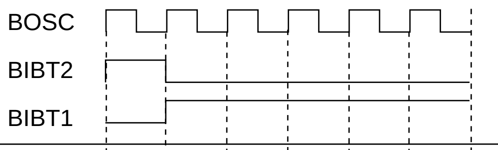
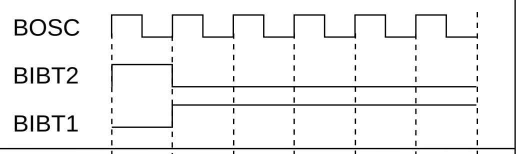
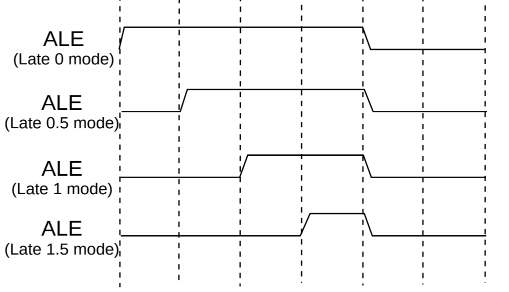
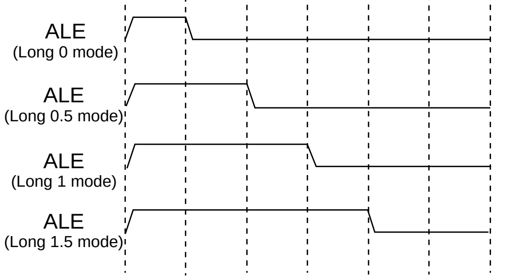
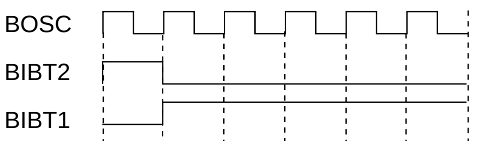
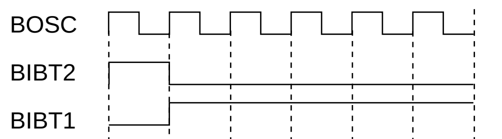
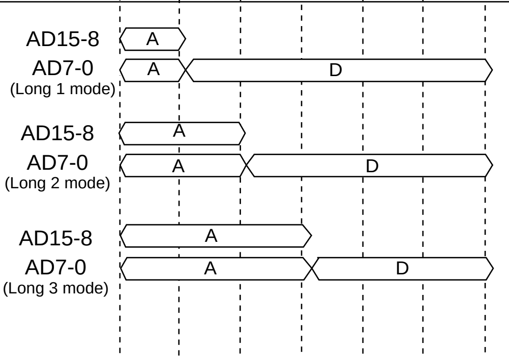
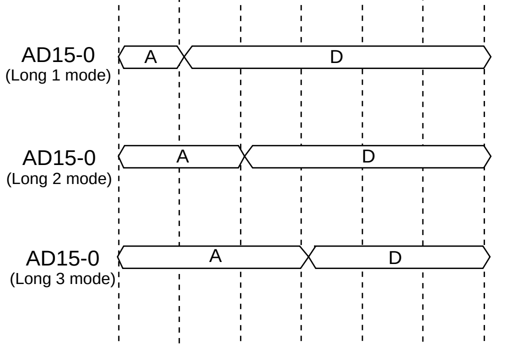
	Late Mode	Long Mode
Base Clock		
	 (The above waveforms are ones in the ALE long 1.5 mode.)	 (The above waveforms are ones in the ALE late 0 mode.)

Table 2-2-7 AD Long Mode (Address/Data Shared Mode)

	8-bit Bus	16-bit Bus
Base Clock		
		

2-3 Handshake Wait Control

2-3-1 Overview

The MN102H55D/55G/F55G controls handshake wait cycles using WAIT pin when reading or writing the data for external memory or other devices.

The MN102H55D/55G/F55G determines the wait cycles using WAIT pin when reading or writing the data. When starting read/write access, input high level to the WAIT pin. High level must be input until BOSC signal falls in T2 interval shown in Figure 2-3-1 because the WAIT pin input level is checked every time BOSC signal falls. While the WAIT pin is high level, the access cycle for the external memory or other devices is continued. On the other hand, when the WAIT pin becomes low level, the access cycle ends 1.5 BOSC cycles later after the next BOSC signal falls.

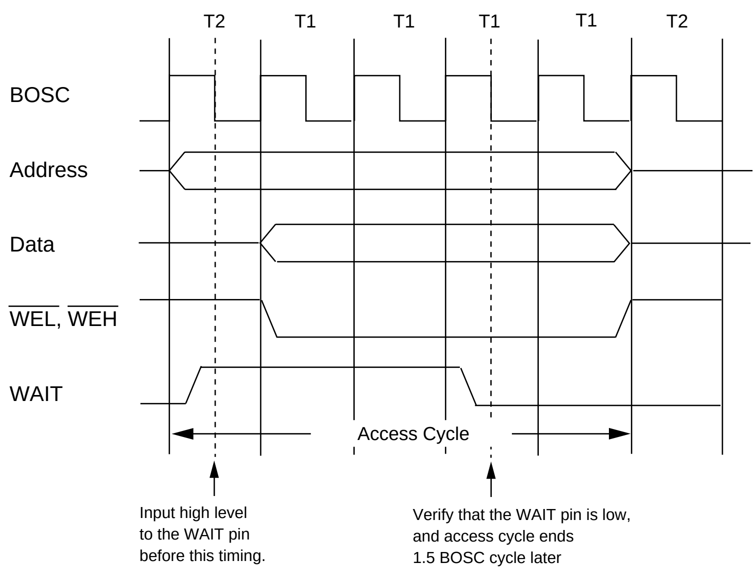


Figure 2-3-1 Handshake Wait Control Timing (1.5 Wait Cycles, Data Write)

When controlling the handshake wait cycles using the WAIT pin, the fixed wait interval can be determined when the read/write access starts. During the fixed wait interval, waits are inserted to read/write access cycles regardless of the WAIT pin status. When the fixed wait cycle ends, the normal handshake wait control using the WAIT pin is selected. This function is available when the handshake access is performed to the external memory or other devices. After low level is input to the WAIT pin to end the access cycle, high level must be input to the WAIT pin until BOSC signal of the next access cycle falls. If this interval is short, timing to input high level cannot be made. In this case, the next access cycle becomes no wait cycle. Using this function prevents such errors.

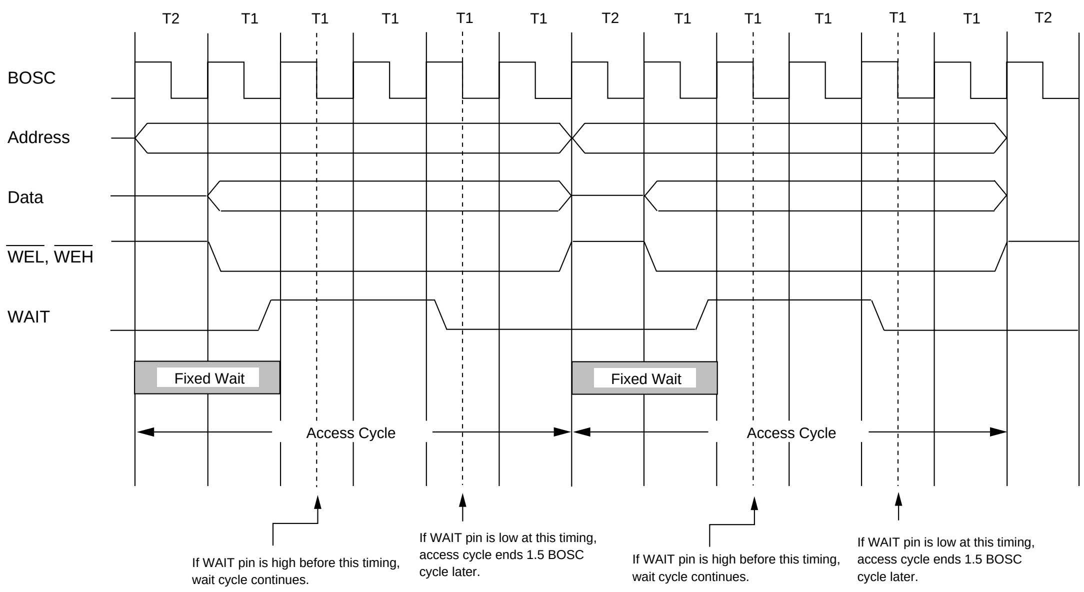


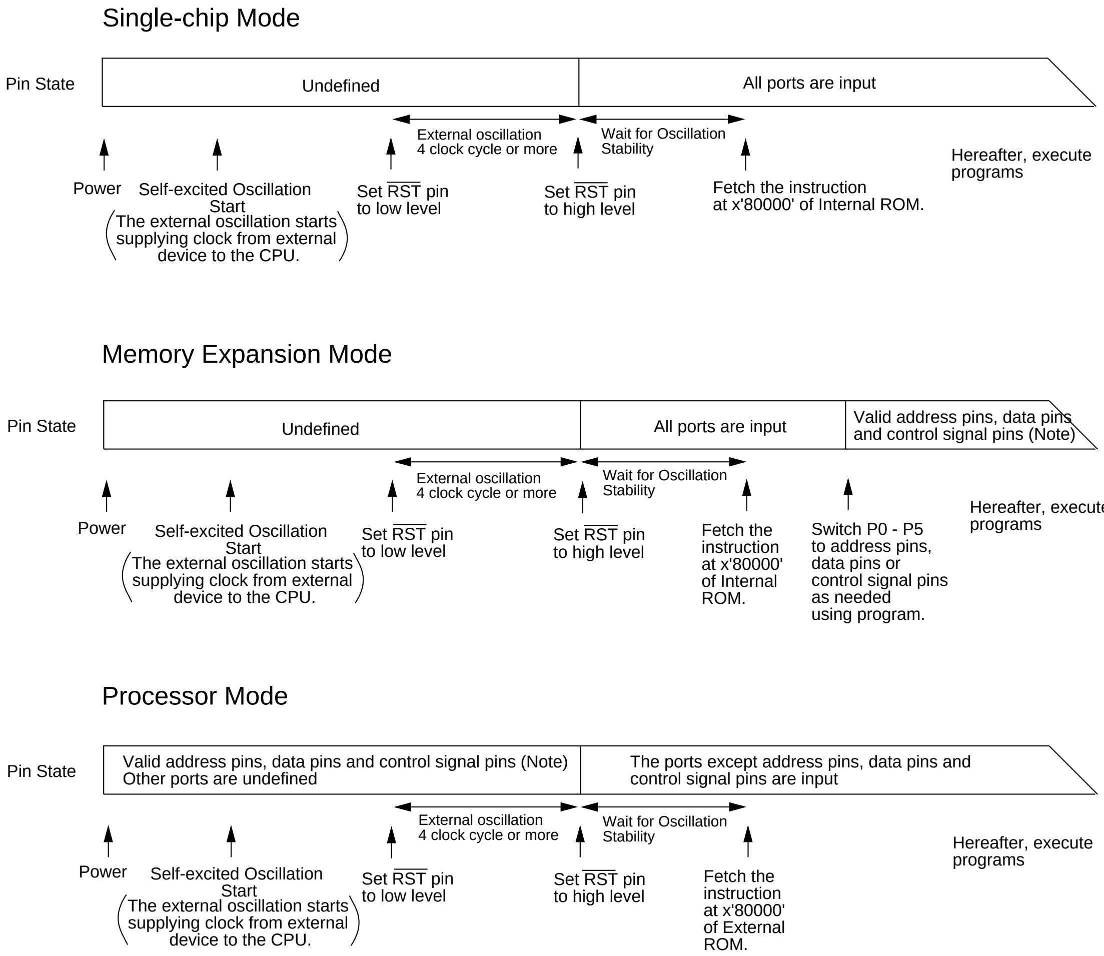
Figure 2-3-2 Fixed Wait and Handshake Wait Control Timing (1 Wait Cycle as Fixed Wait, 2 Wait Cycles as Whole Wait, Data Write)

Set '1111' to bits for the number of wait cycles of the EXWMD register to control handshake wait cycles. When the fixed wait cycle is required, set the necessary number of wait cycles to bits for number of fixed wait cycles of the MEMMD 2 register at the same time. (The wait cycle range from 0 to 3.5 cycles is set in 0.5 wait cycle.)

2-4 Activation Sequence

2-4-1 Activation Sequence of Each Mode

This section describes the activation sequence and the pin state after power turns on in single-chip mode, memory expansion mode and processor mode. The activation sequences and pin states in mask ROM version (MN102H55G) and those in flash version (MN102HF55G) are same.



(Note) The $\overline{\text{CSn}}$ pin, $\overline{\text{RE}}$ pin, $\overline{\text{WEL}}$ pin and $\overline{\text{WEH}}$ pin are high level.
The initial values of the address pins and data pins are undefined.

Figure 2-4-1 Activation Sequence of Each Mode

Chapter 3 Interrupts

3-1 Interrupt Groups

3-1-1 Overview

The most important factor in the real time control is how fast the program moves to the interrupt handler processing. The MN102H55D/55G/F55G improves the interrupt response by aborting instructions, including the multiply and divide instruction, which require multiple clock cycles. The aborted instruction is executed once again after it is returned from the interrupt service routine.

This section describes the overview of the interrupt system. The MN102H55D/55G/F55G contains 56 interrupt groups. Each interrupt group controls interrupts. An interrupt is generated speedily because one interrupt vector is assigned to each interrupt group. Interrupt groups are classified into 14 classes, which set its interrupt level. All interrupts from the peripheral circuits (such as timers) and external pins, except reset interrupts, are registered into interrupt group controller. Once interrupts are registered, interrupt requests are sent to the CPU according to the interrupt priority level (level 0 to 6) set in interrupt group controller. Groups 0 to 3 are interrupts for the system. Table 3-1-1 shows the comparison between this LSI series and the previous 16-bit series.

Table 3-1-1 Comparison of MN102H55D/55G/F55G and MN102B00/MN102L00

Parameters	MN102B00/MN102L00	MN102H55D/55G/F55G
Interrupt Groups (IAGR group numbers)	4 vectors per group (separated by interrupt service routine)	1 vector per group (Generated the group number for each interrupt)
Interrupt Response Time	Good	Excellent
Interrupt Level Setup	4 vectors per level	4 vectors per level
Software Compatibility	———	Easily modified

The MN102H55D/55G/F55G has five external interrupt pins and eight key interrupt pins. The IRQTRG register, the KEYTRG register and the KEYCTR register set the interrupt conditions (positive edge, negative edge, both edges or low level).

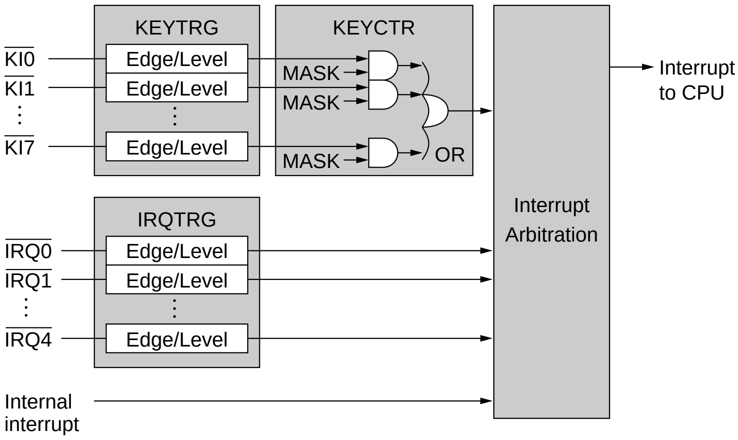
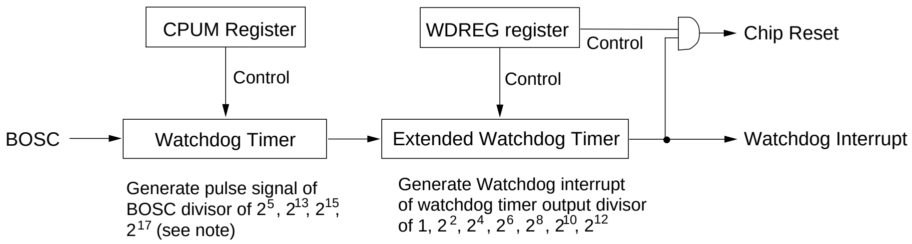


Figure 3-1-1 Interrupt Controller Block Diagram

The MN102H55D/55G/F55G contains the watchdog timer and the extended watchdog timer. The CPUM register and the WDREG register sets the interval until a watchdog interrupt occurs, watchdog timer/extended watchdog timer clear, the chip reset when a watchdog interrupt occurs.



* In the MN102HF55G, use only the BOSC divisor of 2^{17}.

Figure 3-1-2 Watchdog Timer Block Diagram

■ Notices When Using Watchdog Interrupt

The watchdog interrupt is used to detect error operations. Because of this, the CPU normal operation cannot be guaranteed after the watchdog interrupt service routine. Therefore, do not return the old program from the watchdog interrupt service routine. The watchdog interrupt occurs in the following cases.

(1) The program cannot be executed using the normal algorithm due to the infinity loop or error operations.

(2) The CPU hangs up due to the device errors or system errors. (The CPU hangs up the response signal without recognizing during the access to the external device.)

Especially, in case of (2), the instruction in progress enters the interrupt service routine without completing the instruction execution because the CPU terminates the bus cycles forcibly. In addition, the data may not be transferred correctly during the ATC operation. Due to this, the normal program operation cannot be guaranteed even though the program returns from the interrupt service routine.

Table 3-1-2 Interrupt Vector and Class Assignment

MN102H00 CPU Core

$\overline{\text{NMI}}$

Level 0 to 6

Group	Interrupt Vector	Class	Register Address
GROUP 0	Non mascable	CLASS 0	00FC40[R/W]
GROUP 1	Watchdog	CLASS 0	00FC42[R/W]
GROUP 2	Undefined instruction	CLASS 0	00FC44[R/W]
GROUP 3	Error interrupt	CLASS 0	00FC46[R/W]
GROUP 4	Reserved	CLASS 1	-
GROUP 5	Reserved	CLASS 1	-
GROUP 6	Reserved	CLASS 1	-
GROUP 7	Reserved	CLASS 1	-
GROUP 8	External interrupt 0	CLASS 2	00FC50[R/W]
GROUP 9	Timer 0 underflow	CLASS 2	00FC52[R/W]
GROUP 10	Timer 8 underflow	CLASS 2	00FC54[R/W]
GROUP 11	Timer 8 capture A	CLASS 2	00FC56[R/W]
GROUP 12	External interrupt 1	CLASS 3	00FC58[R/W]
GROUP 13	Timer 1 underflow	CLASS 3	00FC5A[R/W]
GROUP 14	Timer 8 capture B	CLASS 3	00FC5C[R/W]
GROUP 15	Timer 9 underflow	CLASS 3	00FC5E[R/W]
GROUP 16	External interrupt 2	CLASS 4	00FC60[R/W]
GROUP 17	Timer 2 underflow	CLASS 4	00FC62[R/W]
GROUP 18	Timer 9 capture A	CLASS 4	00FC64[R/W]
GROUP 19	Timer 9 capture B	CLASS 4	00FC66[R/W]
GROUP 20	External interrupt 3	CLASS 5	00FC68[R/W]
GROUP 21	Timer 3 underflow	CLASS 5	00FC6A[R/W]
GROUP 22	Timer 10 underflow	CLASS 5	00FC6C[R/W]
GROUP 23	Timer 10 capture A	CLASS 5	00FC6E[R/W]
GROUP 24	External interrupt 4	CLASS 6	00FC70[R/W]
GROUP 25	Timer 4 underflow	CLASS 6	00FC72[R/W]
GROUP 26	Timer 10 capture B	CLASS 6	00FC74[R/W]
GROUP 27	Timer 11 underflow	CLASS 6	00FC76[R/W]
GROUP 28	External key interrupt	CLASS 7	00FC78[R/W]
GROUP 29	Timer 5 underflow	CLASS 7	00FC7A[R/W]
GROUP 30	Timer 11 capture A	CLASS 7	00FC7C[R/W]
GROUP 31	Timer 11 capture B	CLASS 7	00FC7E[R/W]
GROUP 32	AD conversion end	CLASS 8	00FC80[R/W]
GROUP 33	Timer 6 underflow	CLASS 8	00FC82[R/W]
GROUP 34	Timer 12 underflow	CLASS 8	00FC84[R/W]
GROUP 35	Timer 12 capture A	CLASS 8	00FC86[R/W]
GROUP 36	Timer 7 underflow	CLASS 9	00FC88[R/W]
GROUP 37	Timer 12 capture B	CLASS 9	00FC8A[R/W]
GROUP 38	Reserved	CLASS 9	-
GROUP 39	Reserved	CLASS 9	-
GROUP 40	Serial 0 transmission end	CLASS 10	00FC90[R/W]
GROUP 41	Serial 0 reception end	CLASS 10	00FC92[R/W]
GROUP 42	Serial 1 transmission end	CLASS 10	00FC94[R/W]
GROUP 43	Serial 1 reception end	CLASS 10	00FC96[R/W]
GROUP 44	Serial 2 transmission end	CLASS 11	00FC98[R/W]
GROUP 45	Serial 2 reception end	CLASS 11	00FC9A[R/W]
GROUP 46	Serial 3 transmission end	CLASS 11	00FC9C[R/W]
GROUP 47	Serial 3 reception end	CLASS 11	00FC9E[R/W]
GROUP 48	Serial 4 transmission end	CLASS 12	00FCA0[R/W]
GROUP 49	Serial 4 reception end	CLASS 12	00FCA2[R/W]
GROUP 50	ETC0 transfer end	CLASS 12	00FCA4[R/W]
GROUP 51	ETC1 transfer end	CLASS 12	00FCA6[R/W]
GROUP 52	ATC0 transfer end	CLASS 13	00FCA8[R/W]
GROUP 53	ATC1 transfer end	CLASS 13	00FCAA[R/W]
GROUP 54	ATC2 transfer end	CLASS 13	00FCAC[R/W]
GROUP 55	ATC3 transfer end	CLASS 13	00FCAE[R/W]

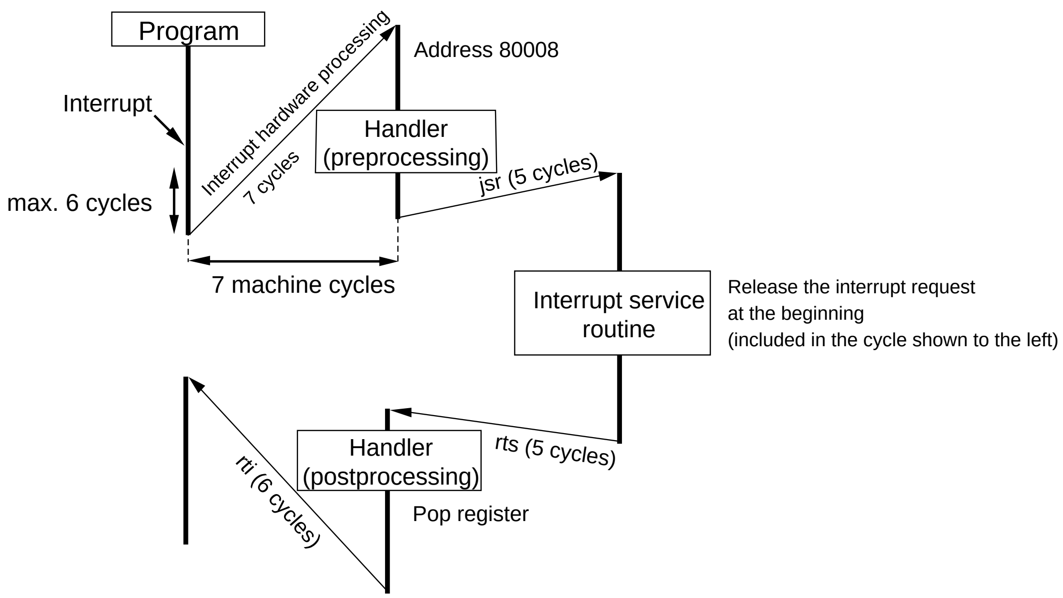


Figure 3-1-3 Interrupt Servicing Time

Table 3-1-3 Handler Preprocessing

Sequence	Assembler	Byte	Cycle

Push register	add -8, A3
	mov A0, (A3)
	movx D0, (4, A3)
		2
		2
		3
			1
			2
			3

Read group number	mov (FC0E), D0	3	1
Generate the first address for interrupt service routine	mov BASE, A0 mov (D0, A0), A0	3 2	1 2
Branch	jsr (A0)	2	5
Total		17	15

Table 3-1-4 Handler Postprocessing

Sequence	Assembler	Byte	Cycle

Pop register	mov (A3), A0
	movx (4, A3), D0
	add 8, A3
		2
		3
		2
			2
			3
			1

Total		7	6

3-1-2 Control Registers

These registers control the interrupt function: the interrupt accept group register (IAGR), the interrupt condition setup register (IRQTRG), the external key interrupt condition setup register (KEYTRG), the external key interrupt enable register (KEYCTR) and the watchdog interrupt extension control register (WDREG).

CLASS 0 (System Interrupt)

7	6	5	4	3	2	1	0
–	–	–	–	–	–	–	ID

Interrupt detect flag
0: Interrupt undetected
1: Interrupt detected

Nonmaskable interrupt	NMICR
Watchdog overflow	WDICR
Undefined instruction	UNICR
Interrupt arbitriation (error interrupt)	EIICR

CLASS 1~13

XnICH

7	6	5	4	3	2	1	0
–		LV		–	–	–	IE

Interrupt priority level
LV[2:0]

Interrupt enable flag (IE)
0: Disable
1: Enable

XnICL

7	6	5	4	3	2	1	0
–	–	–	IR	–	–	–	ID

Interrupt request flag (IR)
0: No interrupt requested
1: Interrupt requested

Interrupt detect flag (ID)
0: Interrupt undetected
1: Interrupt detected

X:	IQ	External interrupt
	TM or T	Timer interrupt
	SC	Serial interrupt
	AT	ATC transfer end interrupt
	AD	AD conversion end interrupt
	KI	Key interrupt

* Some registers do not have LV flags. These bits are read to 0.

The following is an example of setting the interrupt level (LV) and the interrupt enable (IE) in the interrupt control register (XnICH). Interrupts must be disabled during this routine.

```
.....              ;
and 0xf7ff,psw     ;  Clear IE flag of PSW
nop                ;       Inserted to ensure that XnICH is accessible
nop                ;       after clearing IE flag completely
mov d0, (XnICH)    ;  Write LV/IE
or 0x0800, psw     ;  Set IE flag of PSW
```

The program does not need to clear the IE flag of PSW to disable interrupts during interrupt servicing, since IE = 0 unless the IE flag is set. The nop instructions can be any instructions except those which write the IE flag of PSW or LV and IE flags of XnICH register. Two nop instructions are inserted in the example to keep the minimum number of cycles to change the IE flag. More than two nop instructions can be inserted.

IAGR: x'00FC0E'

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	GN5	GN4	GN3	GN2	GN1	GN0	—	—
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

Group number

IRQTRG: x'00FCB0'

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	IQ4 TG1	IQ4 TG0	IQ3 TG1	IQ3 TG0	IQ2 TG1	IQ2 TG0	IQ1 TG1	IQ1 TG0	IQ0 TG1	IQ0 TG0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

IQ0TG[1:0]	Interrupt trigger condition for $\overline{\text{IRQ0}}$ pin
IQ1TG[1:0]	Interrupt trigger condition for $\overline{\text{IRQ1}}$ pin
IQ2TG[1:0]	Interrupt trigger condition for $\overline{\text{IRQ2}}$ pin
IQ3TG[1:0]	Interrupt trigger condition for $\overline{\text{IRQ3}}$ pin
IQ4TG[1:0]	Interrupt trigger condition for $\overline{\text{IRQ4}}$ pin
00	Low level
01	Both edges
10	Negative edge
11	Positive edge

KEYTRG: x'00FCB2'

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	KI7 TG1	KI7 TG0	KI6 TG1	KI6 TG0	KI5 TG1	KI5 TG0	KI4 TG1	KI4 TG0	KI3 TG1	KI3 TG0	KI2 TG1	KI2 TG0	KI1 TG1	KI1 TG0	KI0 TG1	KI0 TG0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

KI0TG[1:0]	Interrupt trigger condition for $\overline{\text{KI0}}$ pin
KI1TG[1:0]	Interrupt trigger condition for $\overline{\text{KI1}}$ pin
KI2TG[1:0]	Interrupt trigger condition for $\overline{\text{KI2}}$ pin
KI3TG[1:0]	Interrupt trigger condition for $\overline{\text{KI3}}$ pin
KI4TG[1:0]	Interrupt trigger condition for $\overline{\text{KI4}}$ pin
KI5TG[1:0]	Interrupt trigger condition for $\overline{\text{KI5}}$ pin
KI6TG[1:0]	Interrupt trigger condition for $\overline{\text{KI6}}$ pin
KI7TG[1:0]	Interrupt trigger condition for $\overline{\text{KI7}}$ pin
00	Low level
01	Both edges
10	Negative edge
11	Positive edge

KEYCTR: x'00FCB4'

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	–	–	–	–	–	–	–	–	KI7 EN	KI6 EN	KI5 EN	KI4 EN	KI3 EN	KI2 EN	KI1 EN	KI0 EN
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

KI0EN	OR interrupt trigger condition for $\overline{\text{KI0}}$ pin
KI1EN	OR interrupt trigger condition for $\overline{\text{KI1}}$ pin
KI2EN	OR interrupt trigger condition for $\overline{\text{KI2}}$ pin
KI3EN	OR interrupt trigger condition for $\overline{\text{KI3}}$ pin
KI4EN	OR interrupt trigger condition for $\overline{\text{KI4}}$ pin
KI5EN	OR interrupt trigger condition for $\overline{\text{KI5}}$ pin
KI6EN	OR interrupt trigger condition for $\overline{\text{KI6}}$ pin
KI7EN	OR interrupt trigger condition for $\overline{\text{KI7}}$ pin
0	Do not set
1	Set

WDREG: x'00FC88'

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WD CLR	–	–	–	–	WD P2	WD P1	WD P0	–	–	–	–	–	–	–	WD RST
Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

WDRST	Reset the chip when watchdog interrupt occurs
0	Disable
1	Enable

WDP[2:0]	Watchdog Interrupt Interval
000	Interval set in CPUM register × 1
001	Interval set in CPUM register × 2^2
010	Interval set in CPUM register × 2^4
011	Interval set in CPUM register × 2^6
100	Interval set in CPUM register × 2^8
110	Interval set in CPUM register × 2^{10}

WDCLR	Extended Watchdog Counter Clear
0	Do not clear
1	Clear

Table 3-1-5 List of Interrupt Control Registers

Register	Address	R/W	Function
IARG	x'00FC0E'	R	Interrupt Accepted Group Number Register
NMICR	x'00FC40'	R/W	Nonmaskable Interrupt Control Register
WDICR	x'00FC42'	R/W	Watchdog Interrupt Control Register
UNICR	x'00FC44'	R/W	Undefined Instruction Interrupt Control Register
EIICR	x'00FC46'	R	Error Interrupt Control Register
IRQTRG	x'00FCB0'	R/W	External Interrupt Condition Setup Register
IQ0ICL	x'00FC50'	R/W	External Interrupt 0 Control Register
IQ0ICH	x'00FC51'	R/W	External Interrupt 0 Control Register
IQ1ICL	x'00FC58'	R/W	External Interrupt 1 Control Register
IQ1ICH	x'00FC59'	R/W	External Interrupt 1 Control Register
IQ2ICL	x'00FC60'	R/W	External Interrupt 2 Control Register
IQ2ICH	x'00FC61'	R/W	External Interrupt 2 Control Register
IQ3ICL	x'00FC68'	R/W	External Interrupt 3 Control Register
IQ3ICH	x'00FC69'	R/W	External Interrupt 3 Control Register
IQ4ICL	x'00FC70'	R/W	External Interrupt 4 Control Register
IQ4ICH	x'00FC71'	R/W	External Interrupt 4 Control Register
TM0UICL	x'00FC52'	R/W	Timer 0 Underflow Interrupt Control Register
TM0UICH	x'00FC53'	R/W	Timer 0 Underflow Interrupt Control Register
TM1UICL	x'00FC5A'	R/W	Timer 1 Underflow Interrupt Control Register
TM1UICH	x'00FC5B'	R/W	Timer 1 Underflow Interrupt Control Register
TM2UICL	x'00FC62'	R/W	Timer 2 Underflow Interrupt Control Register
TM2UICH	x'00FC63'	R/W	Timer 2 Underflow Interrupt Control Register
TM3UICL	x'00FC6A'	R/W	Timer 3 Underflow Interrupt Control Register
TM3UICH	x'00FC6B'	R/W	Timer 3 Underflow Interrupt Control Register
TM4UICL	x'00FC72'	R/W	Timer 4 Underflow Interrupt Control Register
TM4UICH	x'00FC73'	R/W	Timer 4 Underflow Interrupt Control Register
TM5UICL	x'00FC7A'	R/W	Timer 5 Underflow Interrupt Control Register
TM5UICH	x'00FC7B'	R/W	Timer 5 Underflow Interrupt Control Register
TM6UICL	x'00FC82'	R/W	Timer 6 Underflow Interrupt Control Register
TM6UICH	x'00FC83'	R/W	Timer 6 Underflow Interrupt Control Register
TM7UICL	x'00FC88'	R/W	Timer 7 Underflow Interrupt Control Register
TM7UICH	x'00FC89'	R/W	Timer 7 Underflow Interrupt Control Register
TM8UICL	x'00FC54'	R/W	Timer 8 Underflow Interrupt Control Register
TM8UICH	x'00FC55'	R/W	Timer 8 Underflow Interrupt Control Register
TM8AICL	x'00FC56'	R/W	Timer 8 Capture A Interrupt Control Register
TM8AICH	x'00FC57'	R/W	Timer 8 Capture A Interrupt Control Register
TM8BICL	x'00FC5C'	R/W	Timer 8 Capture B Interrupt Control Register
TM8BICH	x'00FC5D'	R/W	Timer 8 Capture B Interrupt Control Register
TM9UICL	x'00FC5E'	R/W	Timer 9 Underflow Interrupt Control Register
TM9UICH	x'00FC5F'	R/W	Timer 9 Underflow Interrupt Control Register
TM9AICL	x'00FC64'	R/W	Timer 9 Capture A Interrupt Control Register
TM9AICH	x'00FC65'	R/W	Timer 9 Capture A Interrupt Control Register
TM9BICL	x'00FC66'	R/W	Timer 9 Capture B Interrupt Control Register
TM9BICH	x'00FC67'	R/W	Timer 9 Capture B Interrupt Control Register

TM10UICL	x'00FC6C'	R/W	Timer 10 Underflow Interrupt Control Register
TM10UICH	x'00FC6D'	R/W	Timer 10 Underflow Interrupt Control Register
TM10AICL	x'00FC6E'	R/W	Timer 10 Capture A Interrupt Control Register
TM10AICH	x'00FC6F'	R/W	Timer 10 Capture A Interrupt Control Register
TM10BICL	x'00FC74'	R/W	Timer 10 Capture B Interrupt Control Register
TM10BICH	x'00FC75'	R/W	Timer 10 Capture B Interrupt Control Register
TM11UICL	x'00FC76'	R/W	Timer 11 Underflow Interrupt Control Register
TM11UICH	x'00FC77'	R/W	Timer 11 Underflow Interrupt Control Register
TM11AICL	x'00FC7C'	R/W	Timer 11 Capture A Interrupt Control Register
TM11AICH	x'00FC7D'	R/W	Timer 11 Capture A Interrupt Control Register
TM11BICL	x'00FC7E'	R/W	Timer 11 Capture B Interrupt Control Register
TM11BICH	x'00FC7F'	R/W	Timer 11 Capture B Interrupt Control Register
TM12UICL	x'00FC84'	R/W	Timer 12 Underflow Interrupt Control Register
TM12UICH	x'00FC85'	R/W	Timer 12 Underflow Interrupt Control Register
TM12AICL	x'00FC86'	R/W	Timer 12 Capture A Interrupt Control Register
TM12AICH	x'00FC87'	R/W	Timer 12 Capture A Interrupt Control Register
TM12BICL	x'00FC8A'	R/W	Timer 12 Capture B Interrupt Control Register
TM12BICH	x'00FC8B'	R/W	Timer 12 Capture B Interrupt Control Register
SC0TICL	x'00FC90'	R/W	Serial 0 Transmission End Interrupt Control Register
SC0TICH	x'00FC91'	R/W	Serial 0 Transmission End Interrupt Control Register
SC0RICL	x'00FC92'	R/W	Serial 0 Reception End Interrupt Control Register
SC0RICH	x'00FC93'	R/W	Serial 0 Reception End Interrupt Control Register
SC1TICL	x'00FC94'	R/W	Serial 1 Transmission End Interrupt Control Register
SC1TICH	x'00FC95'	R/W	Serial 1 Transmission End Interrupt Control Register
SC1RICL	x'00FC96'	R/W	Serial 1 Reception End Interrupt Control Register
SC1RICH	x'00FC97'	R/W	Serial 1 Reception End Interrupt Control Register
SC2TICL	x'00FC98'	R/W	Serial 2 Transmission End Interrupt Control Register
SC2TICH	x'00FC99'	R/W	Serial 2 Transmission End Interrupt Control Register
SC2RICL	x'00FC9A'	R/W	Serial 2 Reception End Interrupt Control Register
SC2RICH	x'00FC9B'	R/W	Serial 2 Reception End Interrupt Control Register
SC3TICL	x'00FC9C'	R/W	Serial 3 Transmission End Interrupt Control Register
SC3TICH	x'00FC9D'	R/W	Serial 3 Transmission End Interrupt Control Register
SC3RICL	x'00FC9E'	R/W	Serial 3 Reception End Interrupt Control Register
SC3RICH	x'00FC9F'	R/W	Serial 3 Reception End Interrupt Control Register
SC4TICL	x'00FCA0'	R/W	Serial 4 Transmission End Interrupt Control Register
SC4TICH	x'00FCA1'	R/W	Serial 4 Transmission End Interrupt Control Register
SC4RICL	x'00FCA2'	R/W	Serial 4 Reception End Interrupt Control Register
SC4RICH	x'00FCA3'	R/W	Serial 4 Reception End Interrupt Control Register
AT0ICL	x'00FCA8'	R/W	ATC 0 Transfer End Interrupt Control Register
AT0ICH	x'00FCA9'	R/W	ATC 0 Transfer End Interrupt Control Register
AT1ICL	x'00FCAA'	R/W	ATC 1 Transfer End Interrupt Control Register
AT1ICH	x'00FCAB'	R/W	ATC 1 Transfer End Interrupt Control Register
AT2ICL	x'00FCAC'	R/W	ATC 2 Transfer End Interrupt Control Register
AT2ICH	x'00FCAD'	R/W	ATC 2 Transfer End Interrupt Control Register
AT3ICL	x'00FCAE'	R/W	ATC 3 Transfer End Interrupt Control Register
AT3ICH	x'00FCAF'	R/W	ATC 3 Transfer End Interrupt Control Register

ETC0ICL	x'00FCA4'	R/W	ETC 0 Transfer End Interrupt Control Register
ETC0ICH	x'00FCA5'	R/W	ETC 0 Transfer End Interrupt Control Register
ETC1ICL	x'00FCA6'	R/W	ETC 1 Transfer End Interrupt Control Register
ETC1ICH	x'00FCA7'	R/W	ETC 1 Transfer End Interrupt Control Register
ADICL	x'00FC80'	R/W	AD Conversion End Interrupt Control Register
ADICH	x'00FC81'	R/W	AD Conversion End Interrupt Control Register
KIICL	x'00FC78'	R/W	External Key Interrupt Control Register
KIICH	x'00FC79'	R/W	External Key Interrupt Control Register
KEYTRQ	x'00FCB2'	R/W	External Key Interrupt Condition Setup Register
KEYCTR	x'00FCB4'	R/W	External Key Interrupt Enable Register
WDREG	x'00FCB6'	R/W	Watchdog Interrupt Extension Control Register

The error interrupt control register does not exist in the hardware. The CPU write 'C' to the IAGR register to indicate that it detected an error interrupt if the interrupt cannot be matched.

All registers except IAGR, IRQTRG, KEYTRG, KEYCTR and WDREG allow only byte-accesses. Use the MOVB instruction to set the data.

3-2 Interrupt Setup Examples

3-2-1 External Pin Interrupt Setup Examples

In this example, an interrupt occurs on the negative edge from the external interrupt pin $\overline{\text{IRQ0}}$ (PA0).

On reset, all bits of the external interrupt condition setup register (IRQTRG) are set to 0 and the IRQ0IR flag of the external interrupt 0 control register (IQ0ICL) is set to 0.

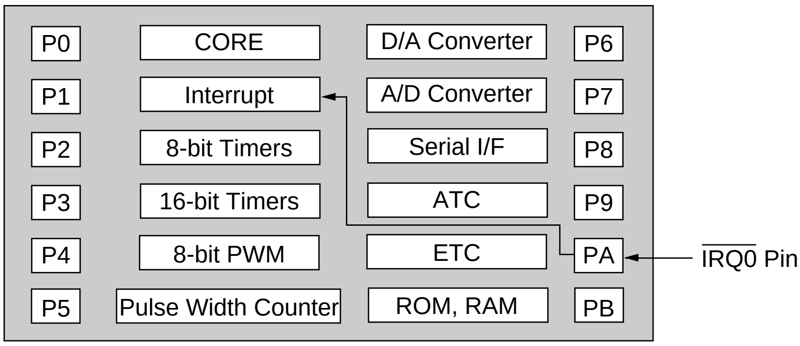


Figure 3-2-1 External Pin Interrupt Block Diagram

■ Interrupt Enable Setting

(1) Set the interrupt conditions for the $\overline{\text{IRQ0}}$ (PA0) pin. In this example, set IQ0TG[1:0] of the IRQTRG register to '2' (bit setting: 10).

IRQTRG: x'00FCB0'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	IQ4 TG1	IQ4 TG0	IQ3 TG1	IQ3 TG0	IQ2 TG1	IQ2 TG0	IQ1 TG1	IQ1 TG0	IQ0 TG1	IQ0 TG0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0

The interrupt level is 5 in this example.

(2) Enable interrupts after clearing all prior interrupt requests. To do this, set the IQ0IR flag of the external interrupt 0 control register (IQ0ICL) to 0, the IQ0LV[2:0] flags of the external interrupt 0 control register (IQ0ICH) to the interrupt level 0 to 6, the IQ0IE flag to '1'.

IQ0ICL: x'00FC50'

7	6	5	4	3	2	1	0
			IQ0 IR				IQ0 ID
			0				0

IQ0ICH: x'00FC51'

7	6	5	4	3	2	1	0
	IQ0 LV2	IQ0 LV1	IQ0 LV0				IQ0 IE
	1	0	1				1

(3) Enable interrupts by writing the IE flag of PSW to 1 and the IMn flag to 7 (bit setting: 111).

Thereafter, an interrupt occurs on the negative edge of the $\overline{\text{IRQ0}}$ (PA0) pin. The program branches to x'080008' when the interrupt is accepted.

■ Interrupt Service Routine

(4) Specify the interrupt group by reading the IAGR register during interrupt preprocessing.

(5) Execute the interrupt service routine.

(6) Clear the IQ0IR flag of the IQ0ICL register.

(7) Return to the main program with the RTI instruction after the interrupt service routine ends.

After the program branches to x'080008', the program generates the interrupt service routine start address and then branches to that address.

During the interrupt service routine, disable an interrupt by setting the IM flag of PSW register to the interrupt level and the IE flag to 0. In addition, other interrupts except nonmaskable interrupts are not accepted unless PSW register is set.

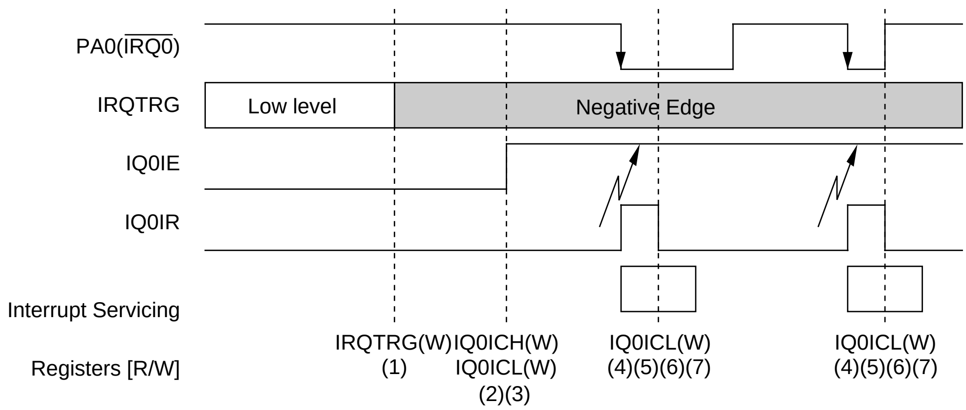


Figure 3-2-2 External Pin Interrupt Timing

3-2-2 Key Input Interrupt Setup Examples

External pins P33 - P30 ($\overline{\text{KI3}}$ - $\overline{\text{KI0}}$) generates key input interrupts. An interrupt signal is generated whenever one of P33 - P30 is low level.

After reset is released, the external key interrupt condition setup register (KEYTRG) is sets low level and the KIIR flag of the external key interrupt control register (KIICL) becomes 0.

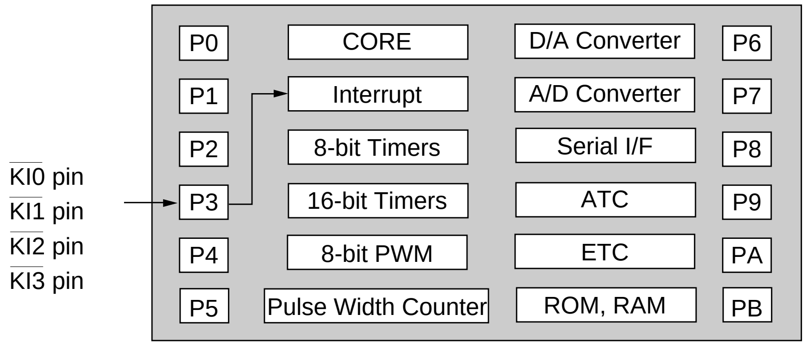


Figure 3-2-3 Key Input Interrupt Block Diagram

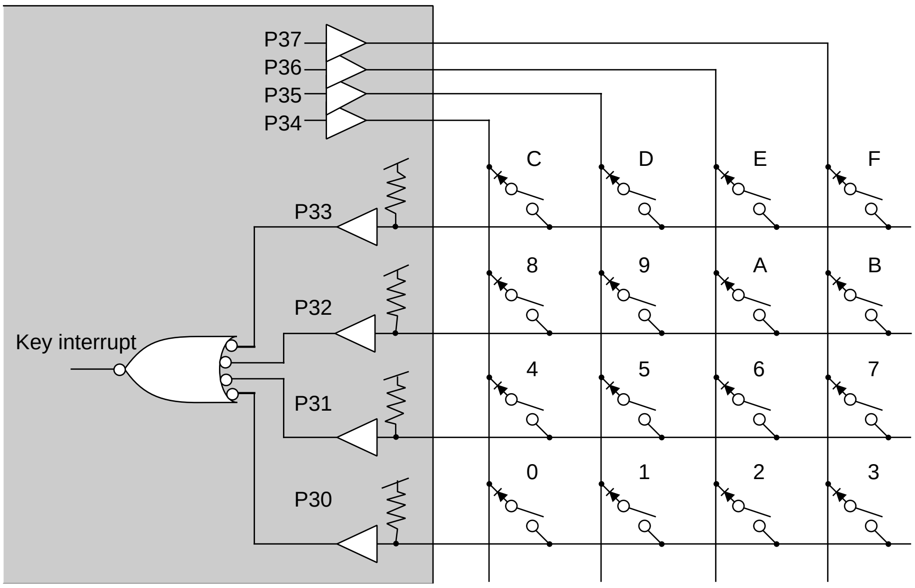


Figure 3-2-4 4×4 Key Matrix Example

■ Interrupt Enable Setting

(1) Set the port functions to the port 3 mode register L and the port 3 mode register H (P3LMD and P3HMD). Set P33 - P30 pins to $\overline{\text{KI}}$ input, select all P37 - P34 pins as ports .

P3LMD: x'00FFF4'

7	6	5	4	3	2	1	0
P3 LMD7	P3 LMD6	P3 LMD5	P3 LMD4	P3 LMD3	P3 LMD2	P3 LMD1	P3 LMD0
1	0	1	0	1	0	1	0

P3HMD: x'00FFF5'

7	6	5	4	3	2	1	0
P3 HMD7	P3 HMD6	P3 HMD5	P3 HMD4	P3 HMD3	P3 HMD2	P3 HMD1	P3 HMD0
0	0	0	0	0	0	0	0

(2) Set the port 3 input/output control register (P3DIR) to the I/O direction. Set P37 - P34 pins to output. P33 - P30 pins are selected as input regardless of the set value of P3DIR because these pins are set to $\overline{\text{KI}}$ input by the port 3 mode register L.

P3DIR: x'00FFE3'

7	6	5	4	3	2	1	0
P3 DIR7	P3 DIR6	P3 DIR5	P3 DIR4	P3 DIR3	P3 DIR2	P3 DIR1	P3 DIR0
1	1	1	1	0	0	0	0

(3) Set P33 - P30 pins to pull-up by the port 3 pull-up control register (P3PUL) not to generate an interrupt when the key is not pushed. Set P37 - P34 pins to output low to generate an interrupt when one of any keys is pushed. Generate a key interrupt signal when any of P33 - P30 pins becomes 0 if one of keys is pushed.

P3PUL: x'00FFB3'

7	6	5	4	3	2	1	0
P3 PLU7	P3 PLU6	P3 PLU5	P3 PLU4	P3 PLU3	P3 PLU2	P3 PLU1	P3 PLU0
0	0	0	0	1	1	1	1

P3OUT: x'00FFC3'

7	6	5	4	3	2	1	0
P3 OUT7	P3 OUT6	P3 OUT5	P3 OUT4	P3 OUT3	P3 OUT2	P3 OUT1	P3 OUT0
0	0	0	0	0	0	0	0

(4) Set the key input pin to low by the KEYTRG register. Enable P33 - P30 key interrupts of the KEYCTR register.

KEYTRG: x'00FCB2'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KI7 TG1	KI7 TG0	KI6 TG1	KI6 TG0	KI5 TG1	KI5 TG0	KI4 TG1	KI4 TG0	KI3 TG1	KI3 TG0	KI2 TG1	KI2 TG0	KI1 TG1	KI1 TG0	KI0 TG1	KI0 TG0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

KEYCTR: x'00FCB4'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
								KI7 EN	KI6 EN	KI5 EN	KI4 EN	KI3 EN	KI2 EN	KI1 EN	KI0 EN
								0	0	0	0	1	1	1	1

When applying to a remote controller, the CPU moves to the STOP mode to reduce power consumption. When an interrupt occurs during the STOP mode, the CPU waits for oscillation stabilization. The CPU waits for up to 4.369 ms with a 30-MHz oscillator. After that, the program branches to x'080008.

(5) Enable interrupts after clearing all prior interrupt requests. To do this, set the KIIR flag of the external key interrupt control register (KIICL) to 0, the KILV2-0 flags of the external key interrupt control register (KIICH) to the interrupt level 0 to 6, and the KIIE flag to '1'.

(6) Enable interrupts by writing the IE flag of PSW to 1 and the IMn flag to 7 (bit setting: 111).

Thereafter, an interrupt occurs when one of any keys is pushed.

■ Interrupt Service Routine

After the program branches to x'080008', the program generates the interrupt service routine start address and then branches to that address.

(7) Specify the interrupt group by reading the IAGR register during interrupt preprocessing.

(8) Execute the key interrupt service routine.

During the interrupt service routine, disable an interrupt by setting the IM flag of PSW register to the interrupt level and the IE flag to 0. In addition, other interrupts except nonmaskable interrupts are not accepted unless PSW register is set.

(9) Clear the KIIR flag of the KIICL register.

(10) Execute the key determination routine.

(11) Return to the main program with the RTI instruction after the interrupt service routine ends.

■ Key Determination Routine

(12) Write x'E0' to the port 3 output register (P3OUT). (bit setting: 11100000 (set 0 to only P34)).

Key determination is performed by reading the port 3 input register (P3IN).

(13) The bit corresponding to the port 3 input register (P3IN) becomes 0 if any one of keys 0, 4, 8, C. Check with the bit test instruction (BTST).

(14) Write x'D0' to the P3OUT register. (bit setting: 11010000, set 0 only to P35)

(15) The bit corresponding to the port 3 input register (P3IN) becomes 0 if any one of keys 1, 5, 9, D. Check with the bit test instruction (BTST).

(16) Write x'B0' to the P3OUT register. (bit setting: 10110000, set 0 only to P36)

(17) The bit corresponding to the port 3 input register (P3IN) becomes 0 if any one of keys 2, 6, A, E. Check with the bit test instruction (BTST).

(18) Write x'70' to the P3OUT register. (bit setting: 01110000, set 0 only in P37)

(19) The bit corresponding to the port 3 input register (P3IN) becomes 0 if any one of keys 3, 7, B, F. Check with the bit test instruction (BTST).

The following figure shows the timing of the key input interrupt.

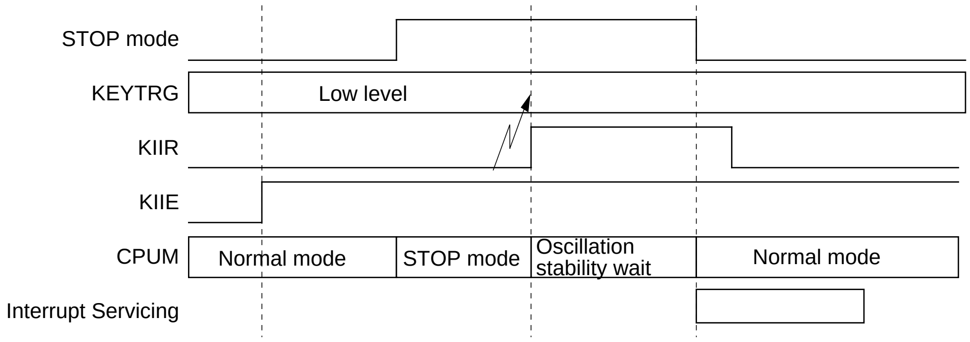


Figure 3-2-5 Key Input Interrupt Timing

3-2-3 Watchdog Timer Interrupt Setup Examples

An interrupt occurs by using the watchdog timer.

The watchdog interrupt is used to detect the CPU errors. The CPU cannot return to the previous operation before the watchdog interrupt occurred after interrupt service routine is executed. Therefore, the CPU must reset after the watchdog interrupt occurred.

The watchdog timer starts by setting the WDRST flag of the CPU mode control register (CPUM) to enable ('0') after reset. When the watchdog timer overflows, a nonmaskable interrupt occurs. This requires to clear the watchdog timer in the main program.

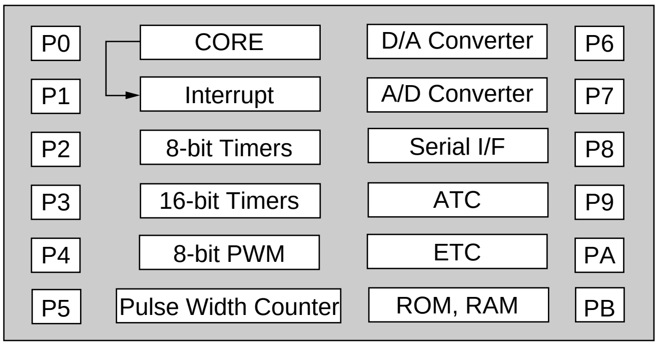


Figure 3-2-6 Watchdog Timer Interrupt Block Diagram

If WDM1 and WDM0 are 00, a watchdog interrupt occurs when the watchdog timer counts 2^{17} BOSC cycles (4.369 ms with a 30-MHz oscillator).
The following is the WDM setting.

00: 2^{17} BOSC cycles
01: 2^{5} BOSC cycles
10: 2^{13} BOSC cycles
11: 2^{15} BOSC cycles

■ Interrupt Enable Setting

(1) Clear the WDRST flag of the CPUM register. This starts the watchdog timer. In addition, set the WDM flags to the time for error detection function.

CPUM: x'00FC00'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WD RST	WD M1	WD M0	-	-	-	-	-	-	-	-	OSC ID	STOP	HALT	OSC1	OSC0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

■ Clearing the Watchdog Timer

Normally, clear the watchdog timer before an interrupt occurs.

(2) Set the WDRST flag of the CPUM register to 1 and then immediately clear it to 0. The watchdog timer clears to 0 when the WDRST flag is 1.

■ Interrupt Service Routine

When an interrupt is generated and accepted, the program branches to x'080008'.

(3) Specify the interrupt group by reading the IAGR register during interrupt preprocessing.

(4) Verify a watchdog interrupt by reading the watchdog interrupt control register (WDICR). Check the WDID flag with the bit test instruction (BTST). If the WDID flag is 1, execute the interrupt service routine.

(5) Clear the WDID flag of the WDICR register.

(6) Return to the main program with the RTI instruction after the interrupt service routine ends.

After the program branches to x'080008', the program generates the interrupt service routine start address and then branches to that address.

During the interrupt service routine, other interrupts are not accepted because IM of PSW becomes the highest level.

The watchdog timer shares the oscillation stabilization wait counter. The WDID flag is cleared to 0 when the program moves to the STOP mode, because the watchdog timer operates as the oscillation stabilization wait counter when the program returns from the STOP mode. The WDID flag is cleared to 0 again after moving to the normal mode. ["2-6 Standby Function" in the MN10200 Series Linear Addressing High-speed Version LSI User Manual]

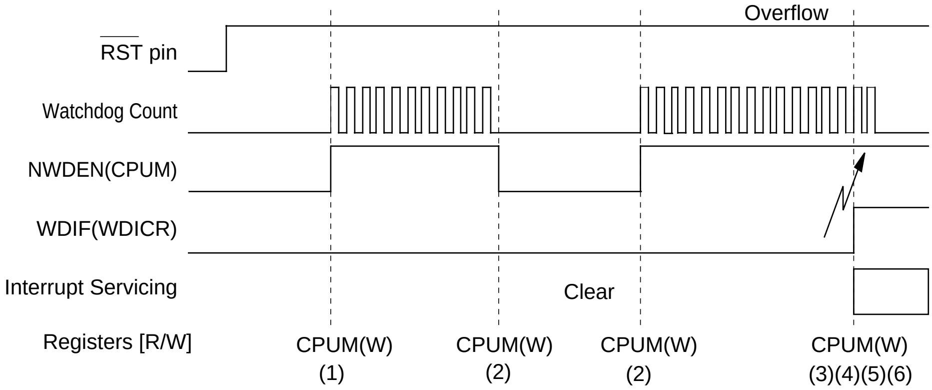


Figure 3-2-7 Watchdog Timer Interrupt Timing

■ Watchdog Timer in STOP Mode

When the watchdog timer is enabled and the CPU switches to STOP mode, the watchdog timer starts counting after it operates as the oscillation stabilization wait counter if the CPU returns to the previous mode (either NORMAL mode or SLOW mode) from STOP mode by an interrupt. In the MN102HF55G (Flash EEPROM version), 2^{17} must be selected as the watchdog interrupt cycle (WDM0='0', WDM1='1') when the CPU moves to STOP mode.

3-2-4 Extended Watchdog Timer Setup Examples

The MN102H55D/55G/F55G has the extended watchdog timer which generates a longer watchdog interrupt than the normal watchdog timer does. In addition, the CPU resets itself instead of generating an interrupt. In this example, if the CPU does not clear the watchdog timer and the extended watchdog timer for 4.47 s with 30-MHz external oscillator, the CPU judges error operation and resets.

The CPU operation when the chip resets using the watchdog timer is the same as the CPU operation when low level is input to $\overline{\mathrm{RST}}$ pin. Generate a pulse signal of 2^{17} BOSC cycles using the CPUM register, and then set 2^{10} using the WDREG register. Since BOSC cycle at 30-MHz external oscillator is approximately 33.3 ns, $33.3\text{ns}\times2^{10}\times2^{17}=4.47$ s and the watchdog timer the extended watchdog timer should be cleared during this interval.

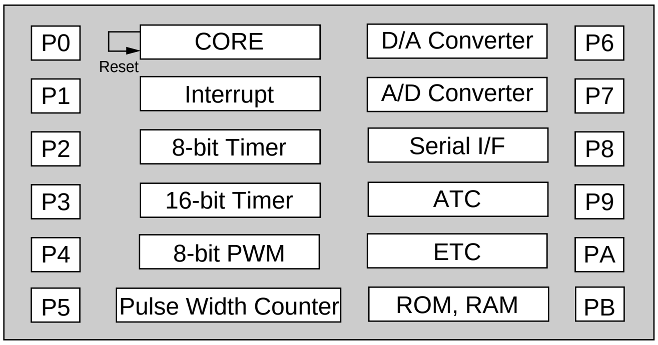


Figure 3-2-8 Extended Watchdog Timer Block Diagram

The following is the WDP[2:0] setting.

000: 1
001: 2^2
010: 2^4
011: 2^6
100: 2^8
101: 2^{10}
110: 2^{12}

■ **Interrupt Enable Setting**

(1) Set WDP[2:0] bits of the WDREG register to the time for error detection function. In this example, set 2^{10}. Since the chip is reset as soon as a watchdog interrupt occurs, set the WDRST flag to 1.

WDREG: x'00FCB6'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WD CLR	-	-	-	-	WD P2	WD P1	WD P0	-	-	-	-	-	-	-	WD RST
0	0	0	0	0	1	0	1	0	0	0	0	0	0	0	1

(2) Clear the WDRST flag of the CPUM register. This starts the watchdog timer and the extended watchdog timer. In addition, set the WDM flags to the time for error detection function. In this example, select 2^{17} .

If WDM1 and WDM0 are 00, a watchdog interrupt occurs when the watchdog timer counts 2^{17} BOSC cycles (4.369 ms with a 30-MHz oscillator).
The following is the WDM setting.

00: 2^{17} BOSC cycles
01: 2^{5} BOSC cycles
10: 2^{13} BOSC cycles
11: 2^{15} BOSC cycles

CPUM: x'00FC00'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WD RST	WD M1	WD M0	-	-	-	-	-	-	-	-	OSC ID	STOP	HALT	OSC1	OSC0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

When the above steps complete, genertae a watchdog interrupt after 4.47 s and reset the chip.

■ Clearing the Watchdog Timer

(3) Set the WDCLR flag of the WDREG register and the WDRST flag of the CPUM register to 1 and then immediately clear them to 0. This clears the watchdog timer and the extended watchdog timer. The watchdog timer is continuously cleared while the WDRST flag is 1 and the extended watchdog timer is continuously cleared while the WDCLR flag is 1. Normally, clear the watchdog timer and the extended watchdog timer before a watchdog interrupt occurs.

Chapter 4 Timers

4

4-1 Summary of 8-bit Timer Functions

4-1-1 Overview

The MN102H55D/55G/F55G contains eight 8-bit down counters that can serve as interval timers, event counters, clock outputs (underflow divided by 2), base clocks for serial interface, or start timing for A/D conversion. The internal clocks (oscillation frequency (BOSC)/2, low-speed frequency (XI)/4) or the external clocks (less than BOSC/4) can be selected as clock sources. Interrupts are generated when timers underflow.

Up to eight 8-bit timers can cascade. For example, cascading timers 4 and 5 forms a 16-bit timer, while cascading timers 0, 1, 2, 3 forms a 32-bit timer. When cascading timers, the clock source of the lowest cascaded timer should be selected as the clock source.
Timers 0 and 4 function as prescalars. They can supply to timers 1 to 3 and timers 5 to 7 as clock sources. This allows low-speed frequency generation and synchronization between timers easily. In addition, they can supply to 16-bit timers as clock sources.

The BOSC frequency is the same as the high-speed oscillation frequency in the normal mode, while the BOSC frequency is the same as the low-speed oscillation frequency in the slow mode. The XI frequency becomes the low-speed oscillation frequency under any modes.

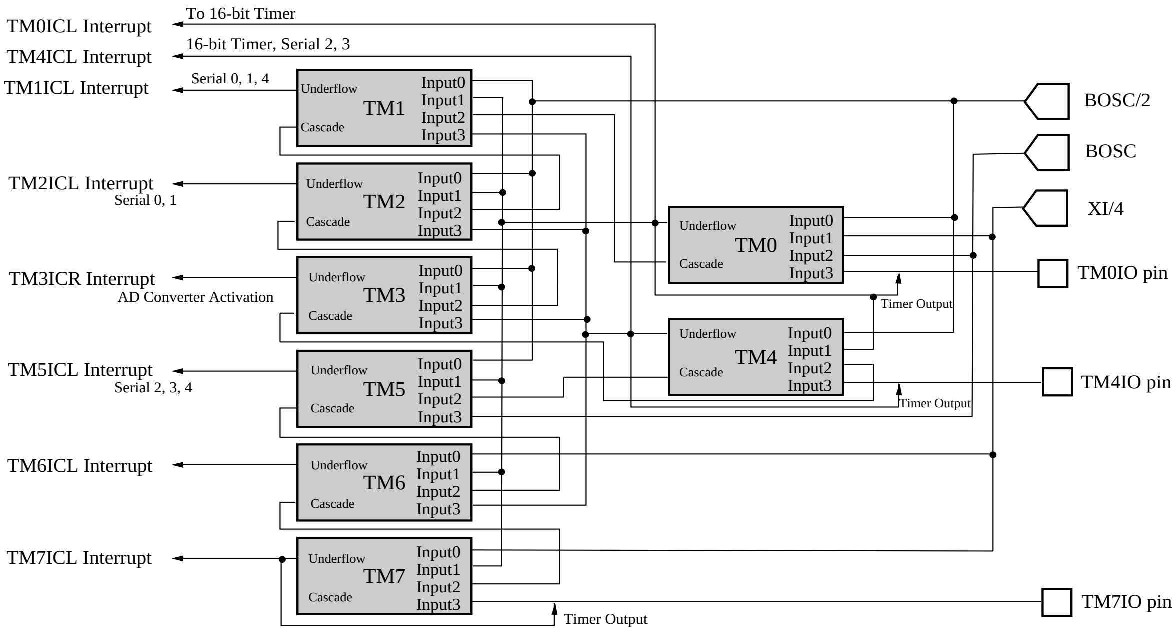


Figure 4-1-1 8-bit Timer Block Diagram

Table 4-1-1 8-bit Timer Functions

		Timer 0	Timer 1	Timer 2	Timer 3	Timer 4	Timer 5	Timer 6	Timer 7
Interrupt Request		TM0ICL	TM1ICL	TM2ICL	TM3ICL	TM4ICL	TM5ICL	TM6ICL	TM7ICL
Interrupt Source		Timer 0 underflow	Timer 1 underflow	Timer 2 underflow	Timer 3 underflow	Timer 4 underflow	Timer 5 underflow	Timer 6 underflow	Timer 7 underflow
Interval Timer		✔	✔	✔	✔	✔	✔	✔	✔
Event Counter		✔	-	-	-	✔	-	-	✔
Clock Source for 16-bit Timer		✔	-	-	-	✔	-	-	-
Timer Output		✔ TM0IO pin	-	-	-	✔ TM4IO pin	-	-	✔ TM7IO pin
Clock Source for Serial Interface		-	✔	✔	-	✔	✔	-	-
A/D Conversion Trigger		-	-	-	✔	-	-	-	-
Clock Sources	0	BOSC/2	BOSC/2	BOSC/2	BOSC/2	BOSC/2	BOSC/2	XI/4	XI/4
	1	XI/4	TM0 underflow	TM0 underflow	TM0 underflow	TM0 underflow	TM0 underflow	TM0 underflow	TM0 underflow
	2	BOSC	Cascade	Cascade	Cascade	Cascade	Cascade	Cascade	Cascade
	3	TM0IO pin	TM4 underflow	TM4 underflow	TM4 underflow	TM4IO pin	BOSC	TM4 underflow	TM7IO pin
Cascade			✔	✔	✔	✔	✔	✔	✔

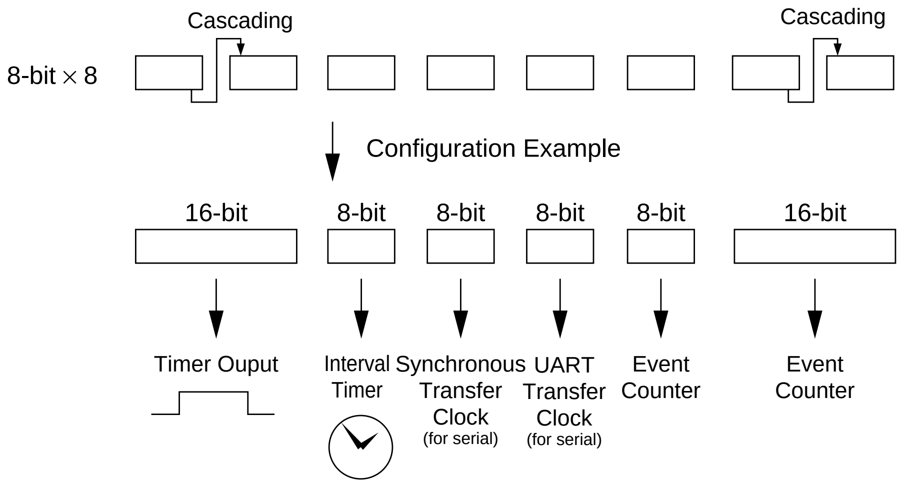


Figure 4-1-2 Timer Configuration

Cascading 8-bit counters forms a 16-bit timer, 24-bit timer, 32-bit timer, 40-bit timer, 48-bit timer, 56-bit timer, or 64-bit timer.

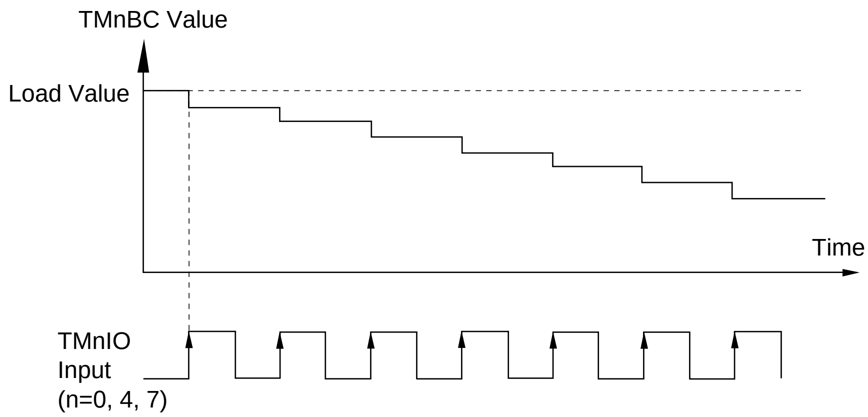


Figure 4-1-3 8-bit Event Counter Input Timing

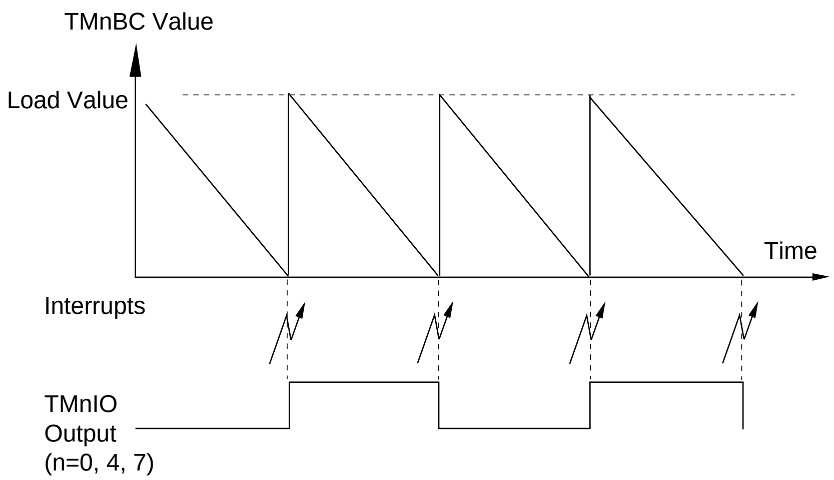


Figure 4-1-4 8-bit Timer Output and Interval Timer Timing

4-1-2 8-bit Timer Control Registers

The timer binary counters (TMnBC), the timer base registers (TMnBR) and the timer mode registers (TMnMD) control timer/counter functions. (n=0 to 7)

TMnBC [R]

	TMn BC7	TMn BC6	TMn BC5	TMn BC4	TMn BC3	TMn BC2	TMn BC1	TMn BC0
Reset	0	0	0	0	0	0	0	0

TMnBR [R/W]

	TMn BR7	TMn BR6	TMn BR5	TMn BR4	TMn BR3	TMn BR2	TMn BR1	TMn BR0
Reset	0	0	0	0	0	0	0	0

TMnMD [R/W]

	TMn EN	TMn LD					TMn S1	TMn S0
Reset	0	0					0	0

TMnS[1:0]	Clock Source Selection
00	BOSC divided by 2, XI divided by 4 *1
01	Timer 0 underflow, XI divided by 4 *1
10	Cascading, BOSC *1
11	Pin input, Timer 4 underflow, BOSC *1

TMnLD	Read TMnBR value to TMnBC
0	No operation
1	Read TMnBR value to TMnBC Reset 2 divisor circuit for timer output

TMnEN	Timer n Counting Operation
0	Counting stop
1	Count operation

*1 Since the settings may differ depending on timers, check each register explanation in Appendix Section.

Table 4-1-2 List of 8-bit Timer Control Registers

Register		Address	R/W	Function
Timer 0	TM0BC TM0BR TM0MD	x'00FE00' x'00FE10' x'00FE20'	R R/W R/W	Timer 0 Binary Counter Timer 0 Base Register Timer 0 Mode Register
Timer 1	TM1BC TM1BR TM1MD	x'00FE01' x'00FE11' x'00FE21"	R R/W R/W	Timer 1 Binary Counter Timer 1 Base Register Timer 1 Mode Register
Timer 2	TM2BC TM2BR TM2MD	x'00FE02' x'00FE12' x'00FE22'	R R/W R/W	Timer 2 Binary Counter Timer 2 Base Register Timer 2 Mode Register
Timer 3	TM3BC TM3BR TM3MD	x'00FE03' x'00FE13' x'00FE23'	R R/W R/W	Timer 3 Binary Counter Timer 3 Base Register Timer 3 Mode Register
Timer 4	TM4BC TM4BR TM4MD	x'00FE04' x'00FE14' x'00FE24'	R R/W R/W	Timer 4 Binary Counter Timer 4 Base Register Timer 4 Mode Register
Timer 5	TM5BC TM5BR TM5MD	x'00FE05' x'00FE15' x'00FE25'	R R/W R/W	Timer 5 Binary Counter Timer 5 Base Register Timer 5 Mode Register
Timer 6	TM6BC TM6BR TM6MD	x'00FE06' x'00FE16' x'00FE26'	R R/W R/W	Timer 6 Binary Counter Timer 6 Base Register Timer 6 Mode Register
Timer 7	TM7BC TM7BR TM7MD	x'00FE07' x'00FE17' x'00FE27'	R R/W R/W	Timer 7 Binary Counter Timer 7 Base Register Timer 7 Mode Register

4-1-3 8-bit Timer Block Diagrams

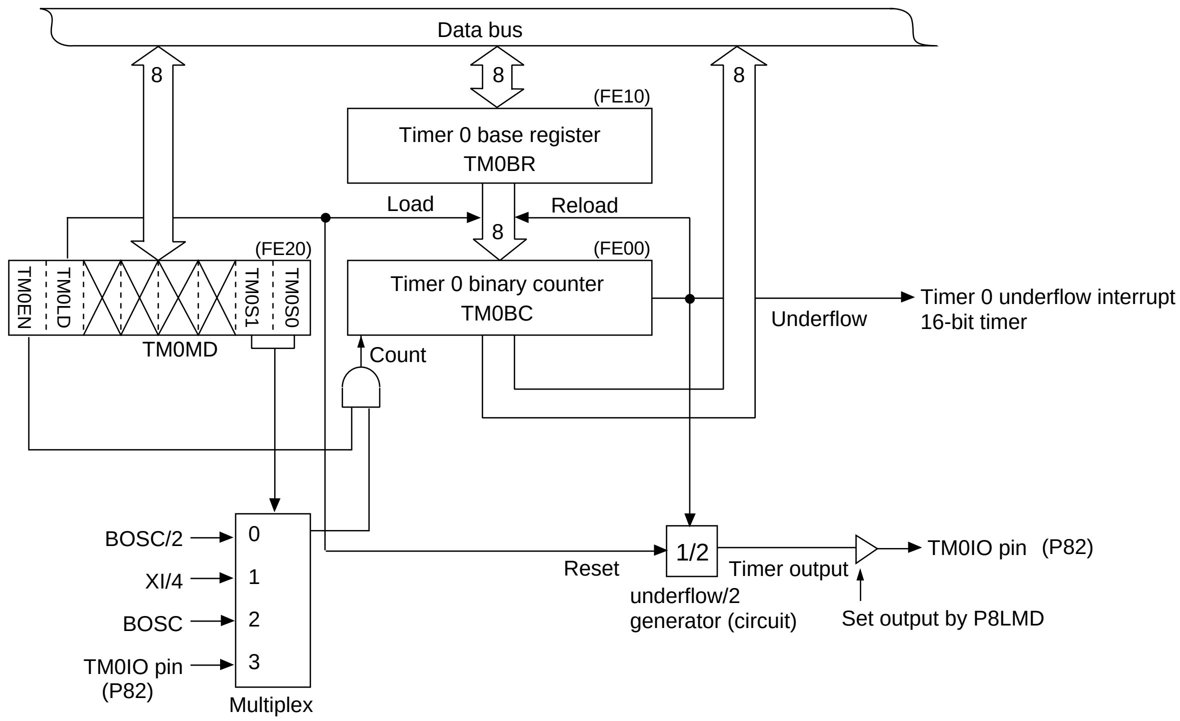


Figure 4-1-5 Timer 0 Block Diagram

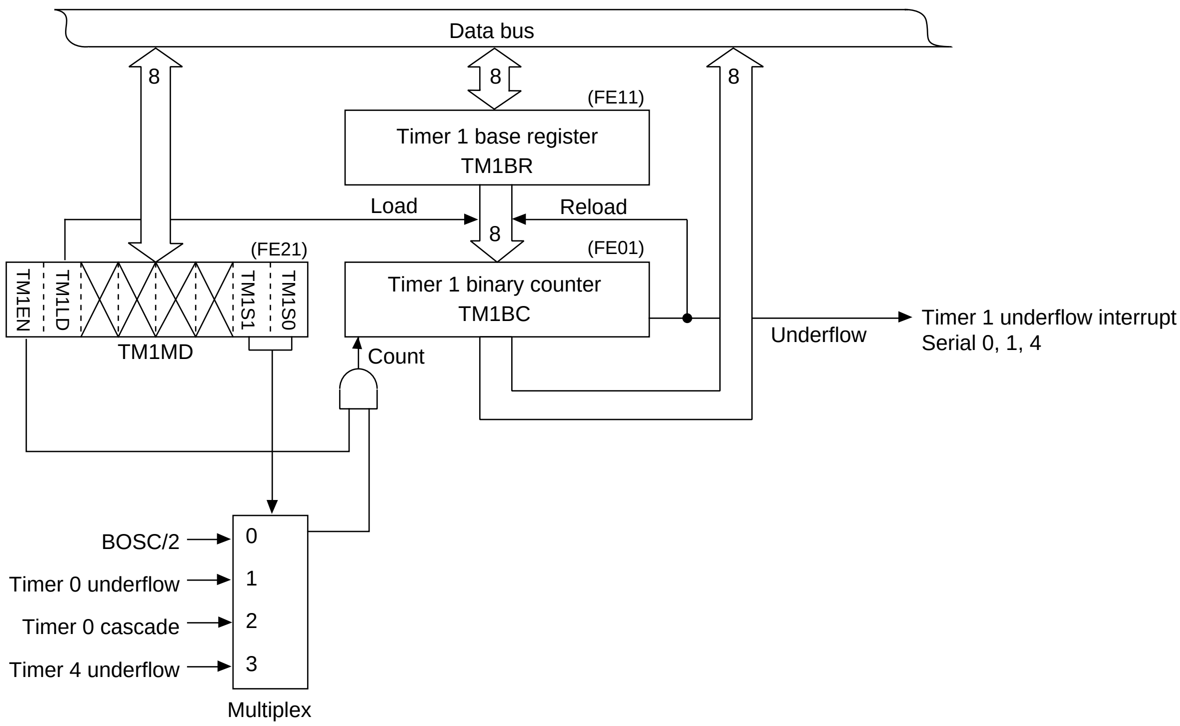


Figure 4-1-6 Timer 1 Block Diagram

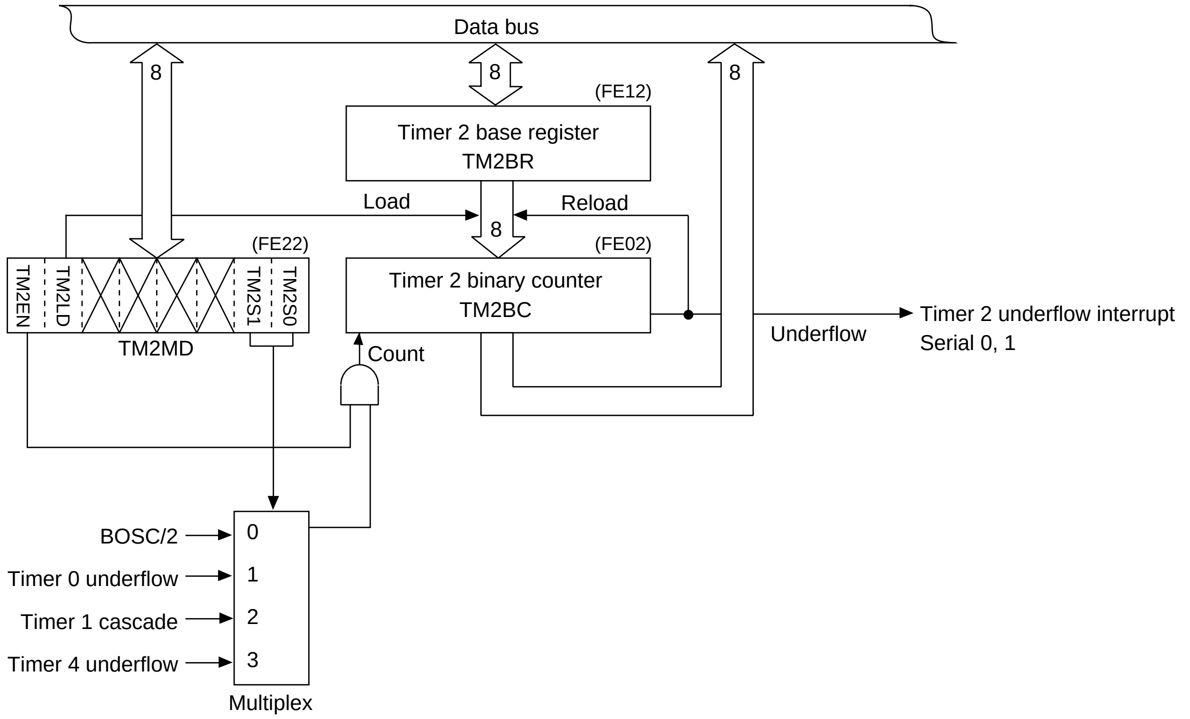


Figure 4-1-7 Timer 2 Block Diagram

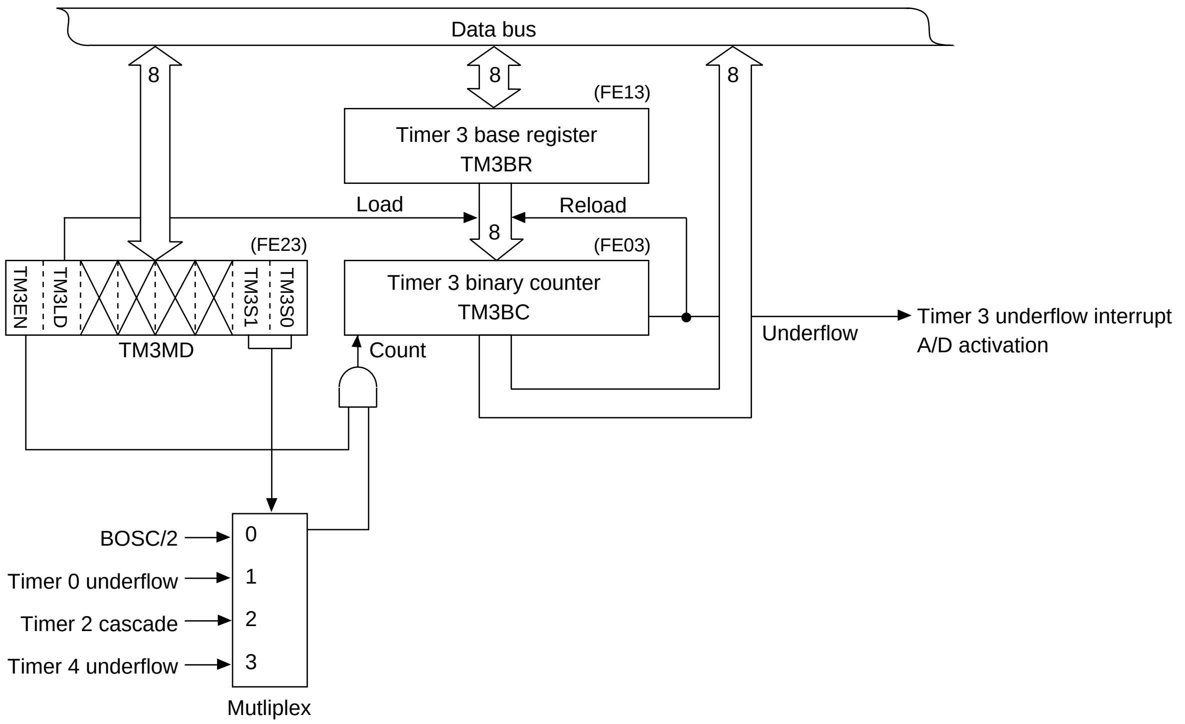


Figure 4-1-8 Timer 3 Block Diagram

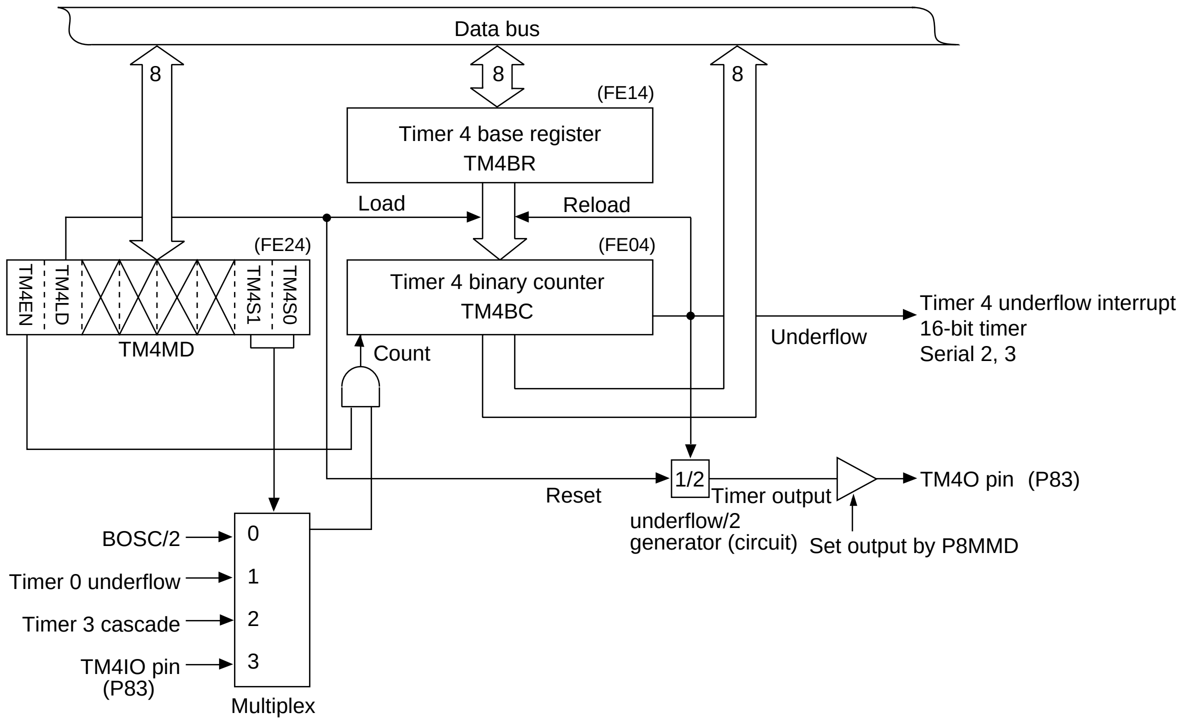


Figure 4-1-9 Timer 4 Block Diagram

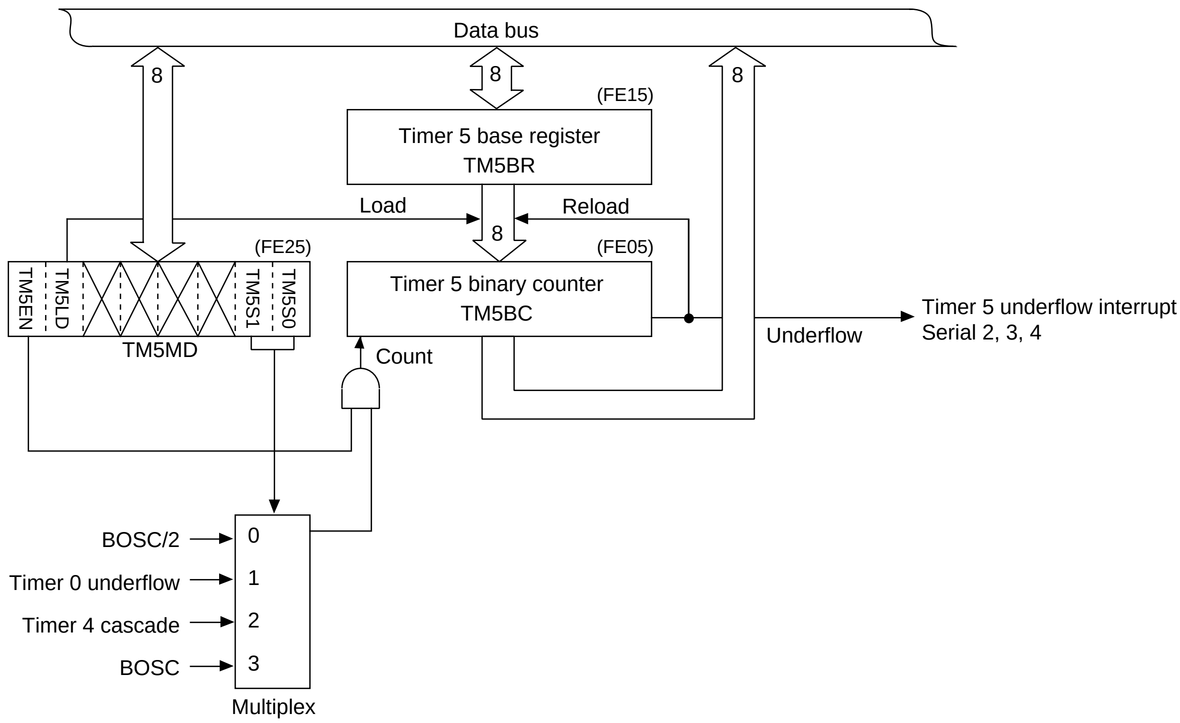


Figure 4-1-10 Timer 5 Block Diagram

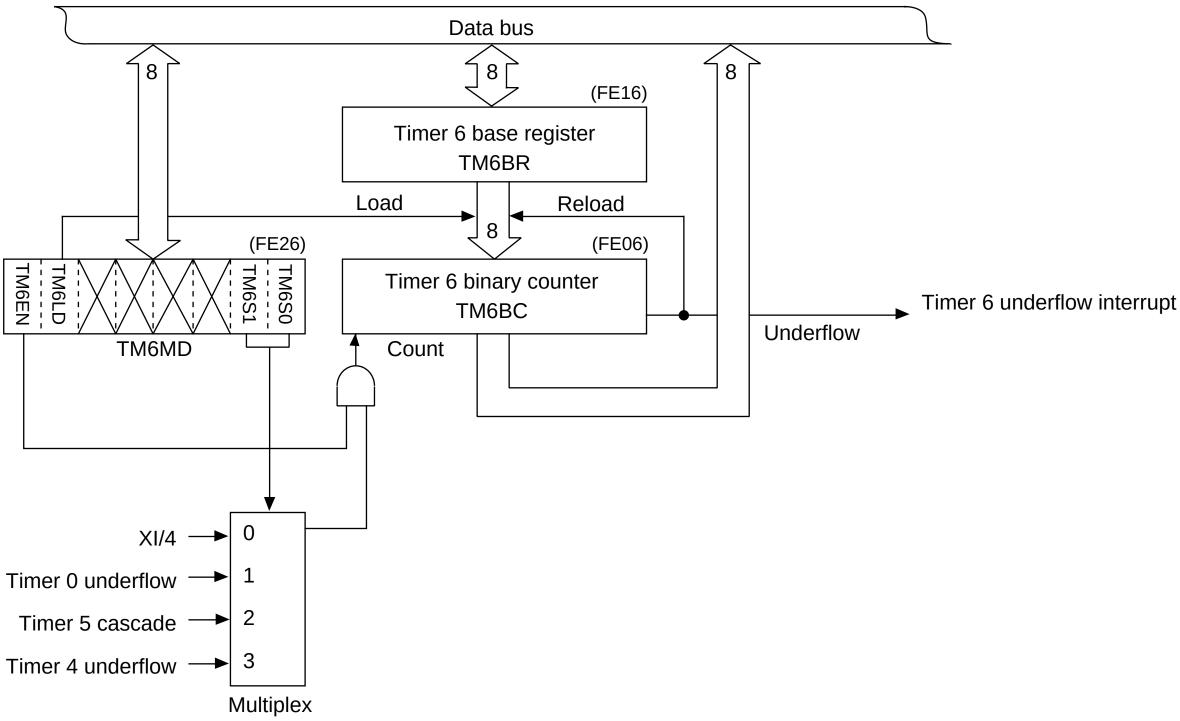


Figure 4-1-11 Timer 6 Block Diagram

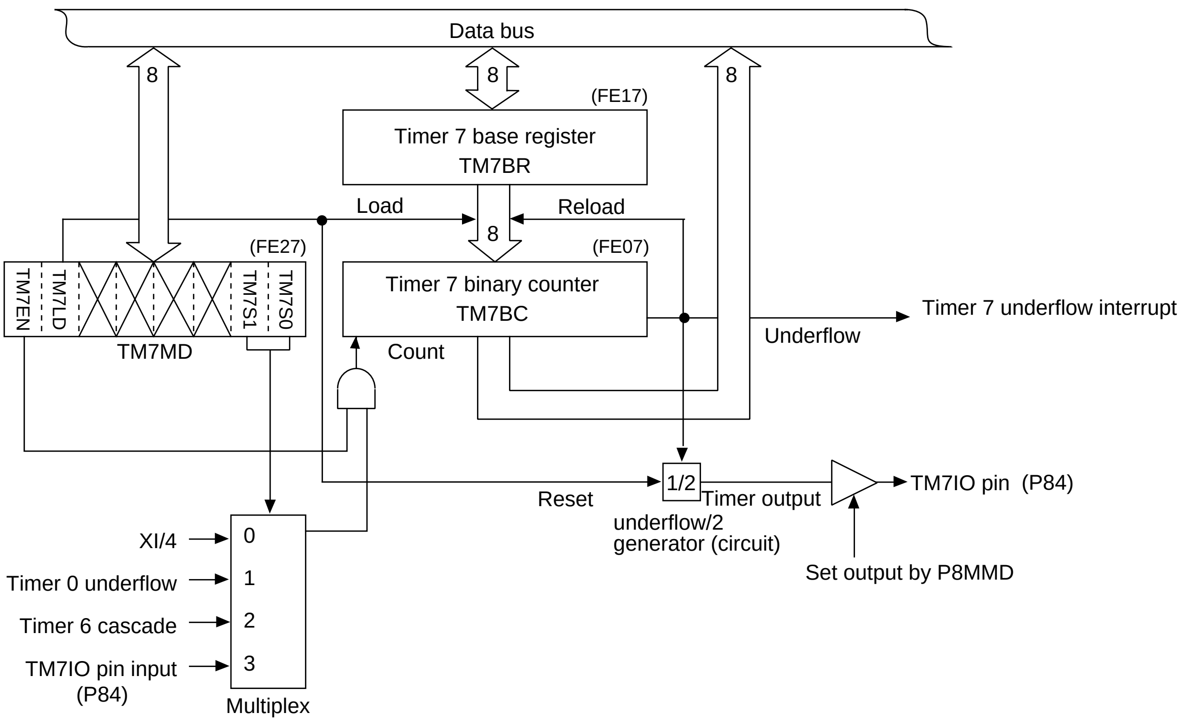


Figure 4-1-12 Timer 7 Block Diagram

4-2 8-bit Timer Setup Examples

4-2-1 Event Counter Using 8-bit Timer

Timer 0 divides TM0IO pin input by 4 and generates an underflow interrupt.

When the pulse is output by the event counter, the change timing is quantized (synchronized with BOSC).

Event counter operates even while the CPU stops. The event counter samples TMnIO pin input on BOSC when the CPU operates. On the other hand, the event counter counts when TMnIO pin input changes during the CPU stop. The CPU transfers to the normal mode after oscillation stability wait when an interrupt is generated. At this point, the event counter counts TMnIO pin input at the change timing until the oscillation stability wait is completed. The event counter, however, starts counting TMnIO pin input at the timing the event counter samples on BOSC.

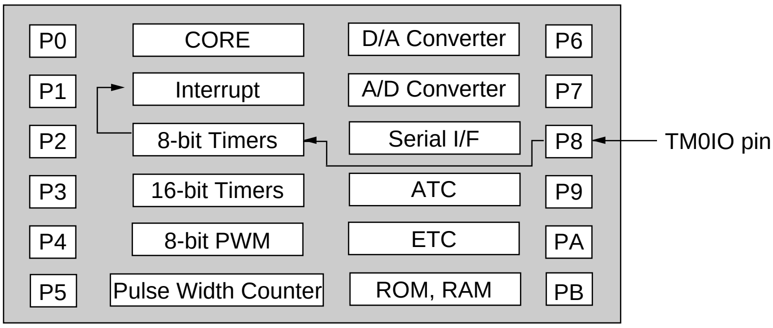


Figure 4-2-1 Event Counter Block Diagram

(1) Set the interrupt enable flag (IE) of the processor status word (PSW) to 1.

(2) Verify that timer 0 counting is stopped with the timer 0 mode register (TM0MD).

This verification is unnecessary after a reset.

TM0MD: x'00FE20'

7	6	5	4	3	2	1	0
TM0 EN	TM0 LD					TM0 S1	TM0 S0
0							

(3) Enable interrupts after clearing all existing interrupt requests. To do this, set IQ0LV[2:0] of the external interrupt 0 control register (IQ0ICH) to interrupt level 0-6, set TM0IR to 0, and set TM0IE to 1. Thereafter, an interrupt will be generated whenever timer 0 underflows.

IQ0ICH/TM0ICL/TM0ICH use only byte access. Use the MOVB instruction.

IQ0ICH: x'00FC50'

7	6	5	4	3	2	1	0
	IQ0 LV2	IQ0 LV1	IQ0 LV0				IQ0 IE
	1	0	0				0

IQ0ICH sets the timer 0 interrupt level. [See "3-1 Interrupt Group"]

The interrupt level is 4 in this example.

TM0ICL: x'00FC52'

7	6	5	4	3	2	1	0
-	-	-	TM0 IR	-	-	-	TM0 ID
			0				0

TM0ICH: x'00FC53'

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM0 IE
							1

(4) Set the timer 0 divisor. Since timer 0 divides TM0IO pin by 4, set the timer 0 base register (TM0BR) to 3. (The valid range for TM0BR is 0 to 255.)

Set the value of timer 0 divisor -1 in the timer 0 base register (TM0BR). If 0 is set in the TM0BR register, the TM0BC value remains 0, but the cycle of the timer 0 underflow and the cycle of the clock source are same.

TM0BR: x'00FE10'

7	6	5	4	3	2	1	0
TM0 BR7	TM0 BR6	TM0 BR5	TM0 BR4	TM0 BR3	TM0 BR2	TM0 BR1	TM0 BR0
0	0	0	0	0	0	1	1

(5) Load TM0BR value to the timer 0 binary counter (TM0BC). At the same time, select TM0IO pin input as the clock source.

Setting TM0EN and TM0LD to 0 is required between (5) and (6) in the bank address version and the linear address version, but this setting is not required in the linear address high-speed version.

TM0MD: x'00FE20'

7	6	5	4	3	2	1	0
TM0 EN	TM0 LD					TM0 S1	TM0 S0
0	1					1	1

(6) Set TM0LD to 0 and TM0EN to 1. This starts the timer. Counting begins at the start of the next cycle.

When the binary counter reaches 0 and loads the value 3 from the base register at the next count, a timer 0 underflow interrupt request will be sent to the CPU.

Do not change the clock source once you have selected it. Selecting the clock source while setting the count operation control will corrupt the value in the binary counter.

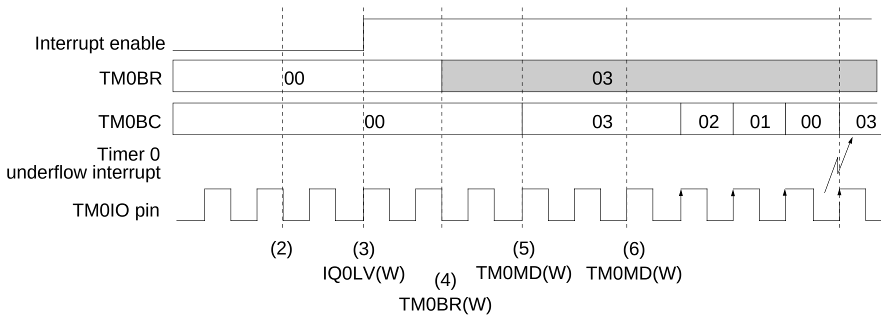


Figure 4-2-2 Event Counter Timing (8-bit Timer)

4-2-2 Clock Output Using 8-bit Timer

Timer 0 and timer 7 output a BOSC/2 divided by 6 (12-cycle) pulse (the duty is 1:1).

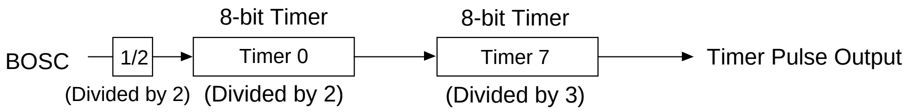


Figure 4-2-3 Clock Output Configuration Example (8-bit Timer)

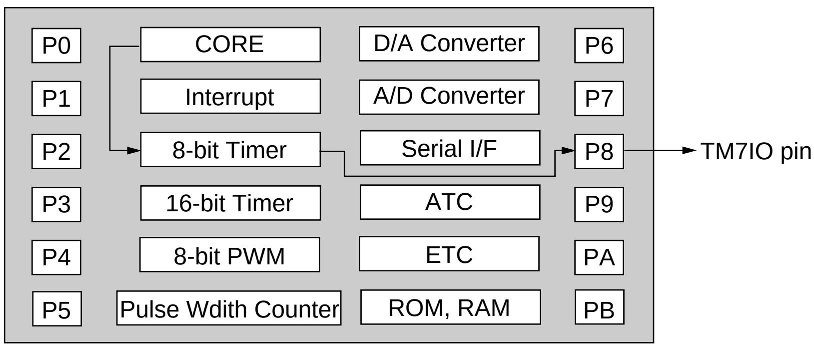


Figure 4-2-4 Clock Output Block Diagram (8-bit Timer)

■ Port Setting

(1) Set TM7IO (P84) of the port 8 to timer output. Set P8MMD[4:2] flags of the port 8 mode control register (P8MMD) to '010' (TM7IO output). With this setting, the direction control of P84 switches to output.

Setting the port 8 I/O control register (P8DIR) is not required. P8DIR operates only when it is used as the port input or output.

P8MMD: x'00FFFD'

7	6	5	4	3	2	1	0
P8 MMD7	P8 MMD6	P8 MMD5	P8 MMD4	P8 MMD3	P8 MMD2	P8 MMD1	P8 MMD0
0	0	0	0	1	0	0	0

■ Timer 0 Setting

This verification is unnecessary after a reset.

(2) Verify that timer 0 counting is stopped with the timer 0 mode register (TM0MD).

TM0MD: x'00FE20'

7	6	5	4	3	2	1	0
TM0 EN	TM0 LD					TM0 S1	TM0 S0
0							

(3) Set the timer 0 divisor. Since timer 0 divides BOSC/2 by 2, set the timer 0 base register (TM0BR) to 1. (The valid range for TM0BR is 0 to 255.)

TM0BR: x'00FE10'

7	6	5	4	3	2	1	0
TM0 BR7	TM0 BR6	TM0 BR5	TM0 BR4	TM0 BR3	TM0 BR2	TM0 BR1	TM0 BR0
0	0	0	0	0	0	0	1

Setting TM0EN and TM0LD to 0 is required between (4) and (5) in the bank address version and the linear address version, but this setting is not required in the linear address high-speed version.

(4) Load TM0BR value to the timer 0 binary counter (TM0BC). At the same time, select BOSC/2 as the clock source.

TM0MD: x'00FE20'

7	6	5	4	3	2	1	0
TM0 EN	TM0 LD					TM0 S1	TM0 S0
0	1					0	0

!

Do not change the clock source once you have selected it. Selecting the clock source while setting the count operation control will corrupt the value in the binary counter.

(5) Set TM0LD and TM0EN of the TM0MD register to 0 and 1 respectively. This starts the timer. Counting begins at the start of the next cycle.

When the binary counter reaches 0 and loads the value 1 from the base register at the next count, a timer 0 underflow interrupt request will be sent to the CPU.

■ Timer 7 Setting

This verification is unnecessary after a reset.

(6) Verify that timer 7 counting is stopped with the timer 7 mode register (TM7MD).

TM7MD: x'00FE27'

7	6	5	4	3	2	1	0
TM7 EN	TM7 LD					TM7 S1	TM7 S0
0							

(7) Set the timer 7 divisor. Since timer 7 divides timer 0 output by 3, set the timer 7 base register (TM7BR) to 2. (The valid range for TM7BR is 0 to 255.)

TM7BR: x'00FE17'

7	6	5	4	3	2	1	0
TM7 BR7	TM7 BR6	TM7 BR5	TM7 BR4	TM7 BR3	TM7 BR2	TM7 BR1	TM7 BR0
0	0	0	0	0	0	1	0

(8) Load TM7BR value to the timer 7 binary counter (TM7BC). At the same time, select the timer 0 underflow as the clock source.

Setting TM0EN and TM0LD to 0 is required between (8) and (9) in the bank address version and the linear address version, but this setting is not required in the linear address high-speed version.

TM7MD: x'00FE27'

7	6	5	4	3	2	1	0
TM7 EN	TM7 LD					TM7 S1	TM7 S0
0	1					0	1

Do not change the clock source once you have selected it. Selecting the clock source while setting the count operation control will corrupt the value in the binary counter.

(9) Set TM7LD to 0 and TM7EN to 1. This starts the timer. Counting begins at the start of the next cycle.

When the timer 7 binary counter (TM7BC) reaches 0 and loads the value 1 from the timer 7 base register (TM7BR) at the next count, the TM7IO output signal is simultaneously inverted. The TM7IO signal starts 0, and then transfer to 1 at the start of the next count cycle. The TM7IO output signal backs to 0 at the start of the following count cycle. By repeating this inversion, the timers generate a 12-cycle (BOSC) clock output signal.

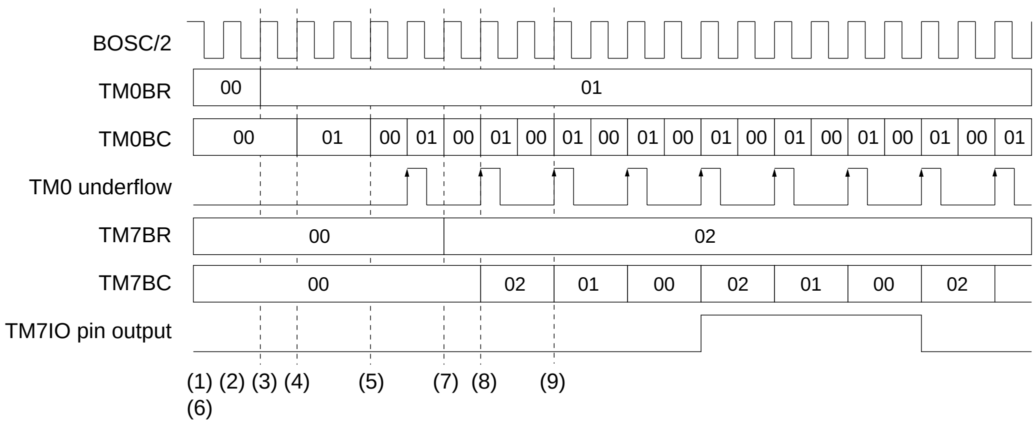


Figure 4-2-5 Clock Output Timing (8-bit Timer)

4-2-3 Interval Timer Using 8-bit Timer

Timer 0, timer 4 and timer 5 divide BOSC/2 by 120,000 and generate an interrupt.

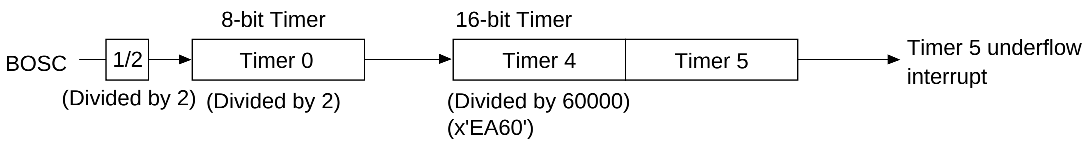


Figure 4-2-6 Interval Timer Configuration Example (8-bit Timer)

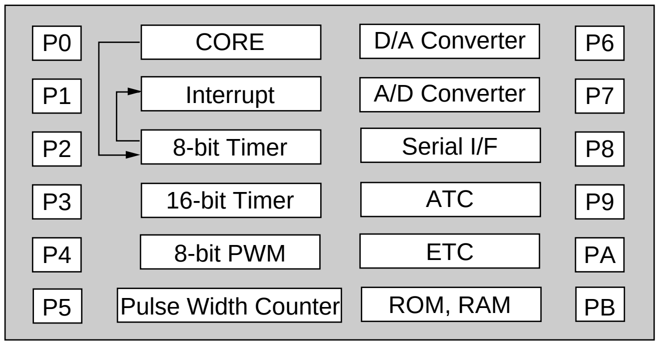


Figure 4-2-7 Interval Timer Block Diagram (8-bit Timer)

■ Timer 0 Setting

This verification is unnecessary after a reset.

(1) Verify that timer 0 counting is stopped with the timer 0 mode register (TM0MD).

TM0MD: x'00FE20'

7	6	5	4	3	2	1	0
TM0 EN	TM0 LD					TM0 S1	TM0 S0
0							

(2) Set the timer 0 divisor. Since timer 0 divides BOSC/2 by 2, set the timer 0 base register (TM0BR) to1. (The valid range for TM0BR is 0 to 255.)

TM0BR: x'00FE10'

7	6	5	4	3	2	1	0
TM0 BR7	TM0 BR6	TM0 BR5	TM0 BR4	TM0 BR3	TM0 BR2	TM0 BR1	TM0 BR0
0	0	0	0	0	0	0	1

(3) Load TM0BR value to the timer 0 binary counter (TM0BC). At the same time, select BOSC/2 as the clock source.

Setting TM0EN and TM0LD to 0 is required between (3) and (4) in the bank address version and the linear address version, but this setting is not required in the linear address high-speed version.

TM0MD: x'00FE20'

7	6	5	4	3	2	1	0
TM0 EN	TM0 LD					TM0 S1	TM0 S0
0	1					0	0

!

Do not change the clock source once you have selected it. Selecting the clock source while setting the count operation control will corrupt the value in the binary counter.

(4) Set TM0LD to 0 and TM0EN to 1. This starts the timer. Counting begins at the start of the next cycle.

When the timer 0 binary counter (TM0BC) reaches 0 and loads the value 1 from the base register at the next count, a timer 0 underflow interrupt request will be sent to the CPU.

■ Timer 4, Timer 5 Settings

(5) Verify that timer counting is stopped with the timer 4 mode register (TM4MD) and the timer 5 mode register (TM5MD).

This verification is unnecessary after a reset.

TM4MD: x'00FE24'

7	6	5	4	3	2	1	0
TM4 EN	TM4 LD					TM4 S1	TM4 S0
0							

TM5MD: x'00FE25'

7	6	5	4	3	2	1	0
TM5 EN	TM5 LD					TM5 S1	TM5 S0
0							

(6) Enable interrupts after clearing all existing interrupt requests. To do this, set KILV[2:0] of the external key interrupt control register (KIICH) to interrupt level 0 to 6, set TM5IR to 0, TM5IE to 1, TM4IR to 0, and TM4IE to 0. Thereafter, an interrupt will be generated whenever timer 5 underflows. The timer 4 underflow is not required.

The timer 5 underflow interrupt level and the external key interrupt level should be the same. The interrupt level is 4 in this example.

KIICH: x'00FC79'

7	6	5	4	3	2	1	0
	KI LV2	KI LV1	KI LV0				KI IE
	1	0	0				

TM4UICL: x'00FC72'

7	6	5	4	3	2	1	0
			TM4U IR				TM4U ID
			0				

TM4UICH: x'00FC73'

7	6	5	4	3	2	1	0
							TM4U IE
							0

TM5UICL: x'00FC7A'

7	6	5	4	3	2	1	0
			TM5U IR				TM5U ID
			0				

TM5UICH: x'00FC7B'

7	6	5	4	3	2	1	0
							TM5U IE
							1

!

When cascading timers, set the lowest timer divisor -1 to the lowest timer base register.

(7) Set the timer divisor. Since timer divides timer output by 60,000 (x'EA60'), set the timer 4 base register (TM4BR) and the timer 5 base register (TM5BR) to x'5F' and x'EA' respectively. (The valid range is 0 to 255.)

TM4BR: x'00FE14'

7	6	5	4	3	2	1	0
TM4 BR7	TM4 BR6	TM4 BR5	TM4 BR4	TM4 BR3	TM4 BR2	TM4 BR1	TM4 BR0
0	1	0	1	1	1	1	1

TM5BR: x'00FE15'

7	6	5	4	3	2	1	0
TM5 BR7	TM5 BR6	TM5 BR5	TM5 BR4	TM5 BR3	TM5 BR2	TM5 BR1	TM5 BR0
1	1	1	0	1	0	1	0

(8) Load TM4BR value and TM5BR to the timer 4 binary counter (TM4BC) and the timer 5 binary counter (TM5BC) respectively. At the same time, select the timer 0 underflow and the timer 4 cascade as the clock source for timer 4 and timer 5 respectively.

Setting TM0EN and TM0LD to 0 is required between (8) and (9) in the bank address version and the linear address version, but this setting is not required in the linear address high-speed version.

TM4MD: x'00FE24'

7	6	5	4	3	2	1	0
TM4 EN	TM4 LD					TM4 S1	TM4 S0
0	1					0	1

TM5MD: x'00FE25'

7	6	5	4	3	2	1	0
TM5 EN	TM5 LD					TM5 S1	TM5 S0
0	1					1	0

Do not change the clock source once you have selected it. Selecting the clock source while setting the count operation control will corrupt the value in the binary counter.

(9) Set TM5LD to 0, TM5EN to 1, TM4LD to 0 and TM4EN to 1. This starts the timer. Counting begins at the start of the next cycle.

When both TM4BC value and TM5BC value reach 0 and the values from TM4BR register and TM5BR register are loaded at the next count, a timer 5 underflow interrupt request will be sent to the CPU. The timer 4 underflow interrupt request can not be used.

When starting the timer, use the MOV instruction to set TM5MD and TM4MD and only use 16-bit write operations. Or set TM5MD first and then set TM4MD.

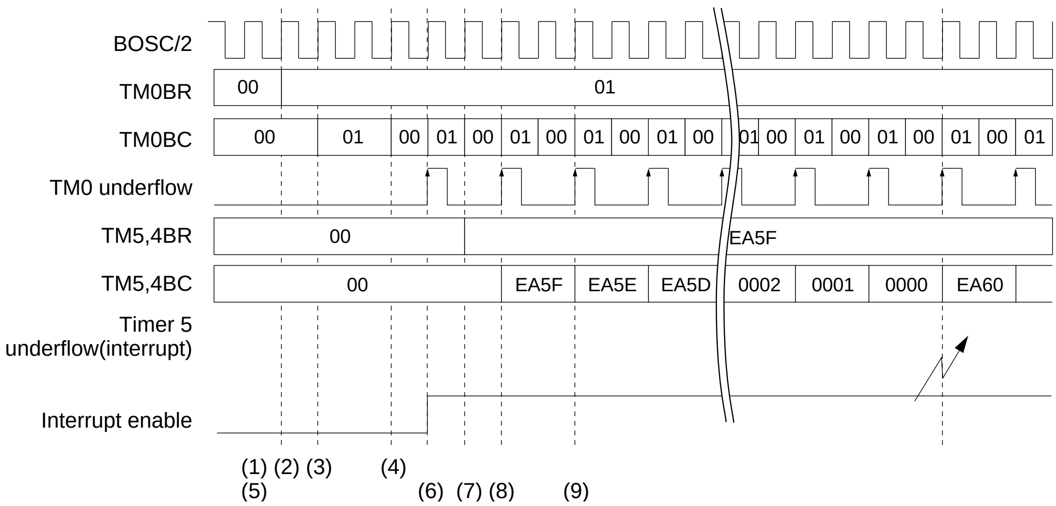


Figure 4-2-8 Interval Timer Timing (8-bit Timer)

4-3 Summary of 16-bit Timer Functions

4-3-1 Overview

The MN102H55D/55G/F55G has five 16-bit up/down counters. Each counter has two compare/capture registers which capture and compare the up/down counter value, generate PWM and interrupts. The PWM has a mode that changes cycle and transition at the beginning of the next cycle. This prevents PWM losses and waveform distortion.

16-bit timer underflow interrupts occur only during down counting.

These counters can serve as interval timers, event counters (in clock oscillation mode), one-phase PWMs, two-phase PWMs, two input captures, two-phase encoders (1x and 4x), one-shot pulse generators, and external count direction controllers. They select internal clocks, external pins, timer 0 underflow or timer 4 underflow as their clock sources.

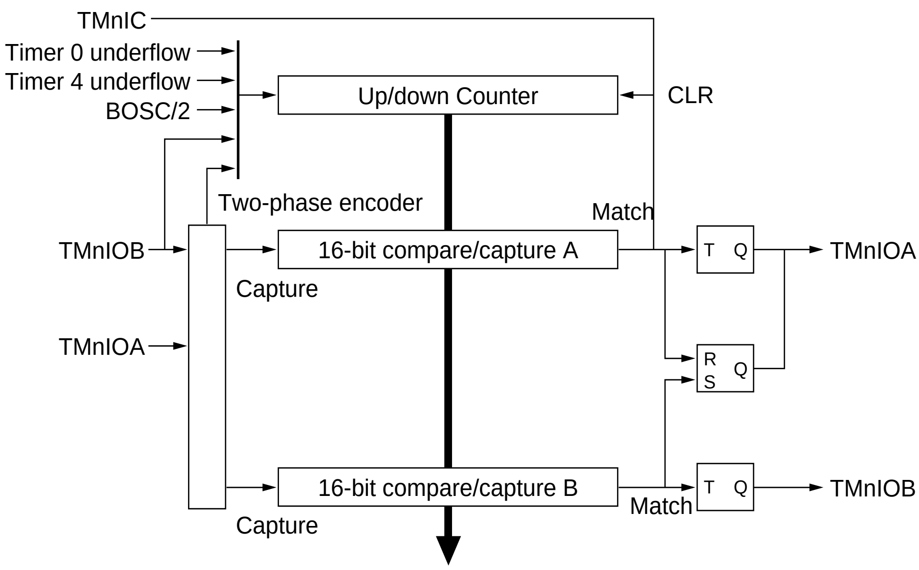


Figure 4-3-1 16-bit Timer Block Diagram

Table 4-3-1 16-bit Timer Functions

	Timer 8	Timer 9	Timer 10	Timer 11	Timer 12
Interrupt Requests	T8UICL T8AICL T8BICL	T9UICL T9AICL T9BICL	T10UICL T10AICL T10BICL	T11UICL T11AICL T11BICL	T12UICL T12AICL T12BICL
Interrupt Sources	• Timer 8 underflow • Timer 8 capture A • Timer 8 capture B	• Timer 9 underflow • Timer 9 capture A • Timer 9 capture B	• Timer 10 underflow • Timer 10 capture A • Timer 10 capture B	• Timer 11 underflow • Timer 11 capture A • Timer 11 capture B	• Timer 12 underflow • Timer 12 capture A • Timer 12 capture B
Clock Sources	• Timer 0 underflow • Timer 4 underflow • TM8IB pin • 1/2 of BOSC • Two-phase encoder of TM8IOA, TM8IOB (4x) • Two-phase encoder of TM8IOA, TM8IOB (1x) • TM8IC pin	• Timer 0 underflow • Timer 4 underflow • TM9IOB pin • 1/2 of BOSC • Two-phase encoder of TM9IOA, TM9IOB (4x) • Two-phase encoder of TM9IOA, TM9IOB (1x)	• Timer 0 underflow • Timer 4 underflow • TM10IOB pin • 1/2 of BOSC • Two-phase encoder of TM10IOA, TM10IOB (4x) • Two-phase encoder of TM10IOA, TM10IOB (1x)	• Timer 0 underflow • Timer 4 underflow • TM11IOB pin • 1/2 of BOSC • Two-phase encoder of TM11IOA, TM11IOB (4x) • Two-phase encoder of TM11IOA, TM11IOB (1x)	• Timer 0 underflow • Timer 4 underflow • TM12IOB pin • 1/2 of BOSC • Two-phase encoder of TM12IOA, TM12IOB (4x) • Two-phase encoder of TM12IOA, TM12IOB (1x)
Count Direction	Up/Down	Up/Down	Up/Down	Up/Down	Up/Down
Interval Timer	✓	✓	✓	✓	✓
Event Counter	✓	✓	✓	✓	✓
PWM	✓	✓	✓	✓	✓
One-shot Pulse Output	✓	✓	✓	✓	✓
1-phase Capture Input	✓	✓	✓	✓	✓
2-phase Capture Input	✓	✓	✓	✓	✓
2-phase Encoder (4x)	✓	✓	✓	✓	✓
2-phase Encoder (1x)	✓	✓	✓	✓	✓
External Count Direction Control	✓	✓	✓	✓	✓
Other	• Switch edge polarity of TM8IC pin input • Switch polarity of TM8IOA,TM8IOB output	• Switch edge polarity of TM9IOB pin input • Clear BC with TM9IOB pin input (Pulse phase difference detection)	• Switch edge polarity of TM10IOB pin input (rising edge, falling edge, or both edges)		

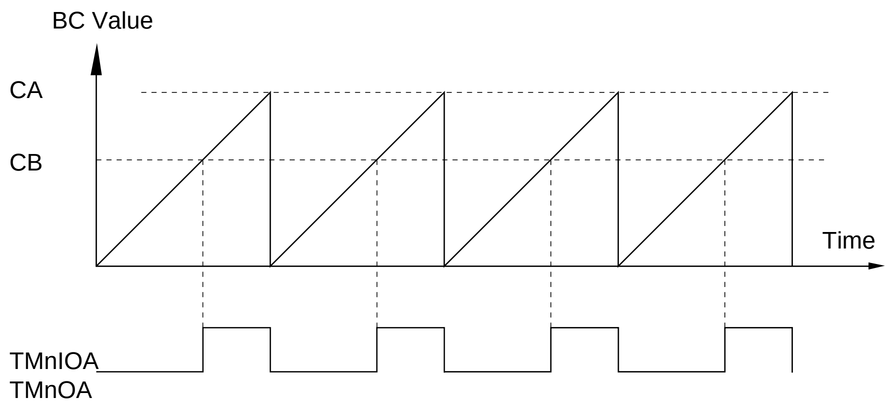


Figure 4-3-2 One-phase PWM Output Timing

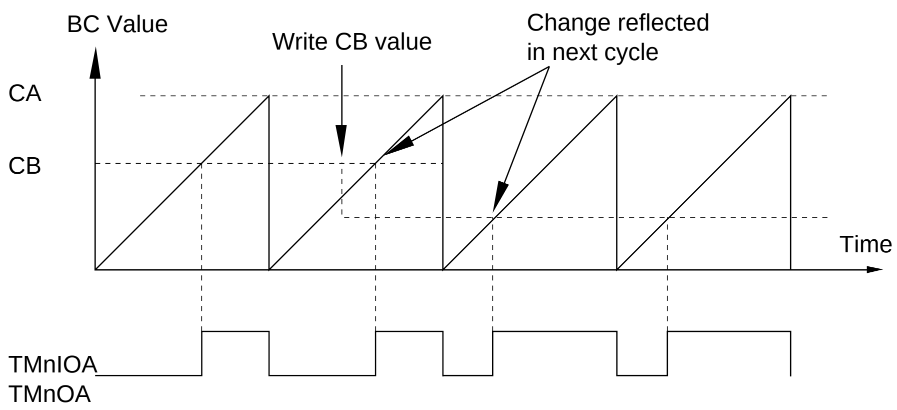


Figure 4-3-3 One-phase PWM Output Timing (with Data Rewrite)

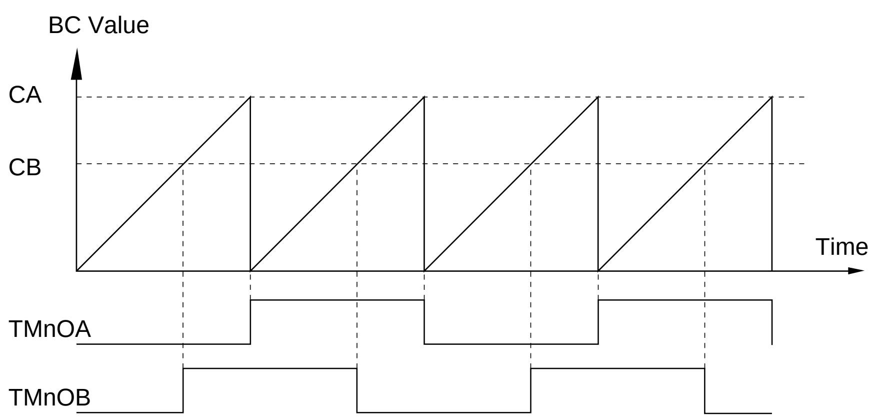


Figure 4-3-4 Two-phase PWM Output Timing

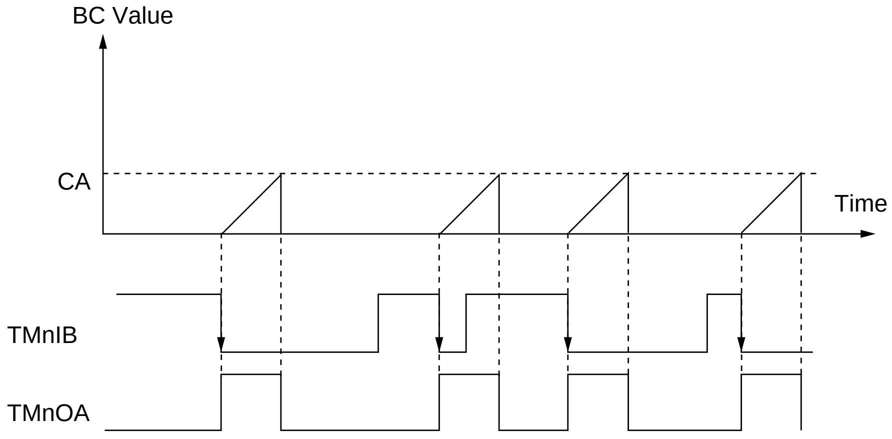


Figure 4-3-5 One-shot Pulse Output Timing

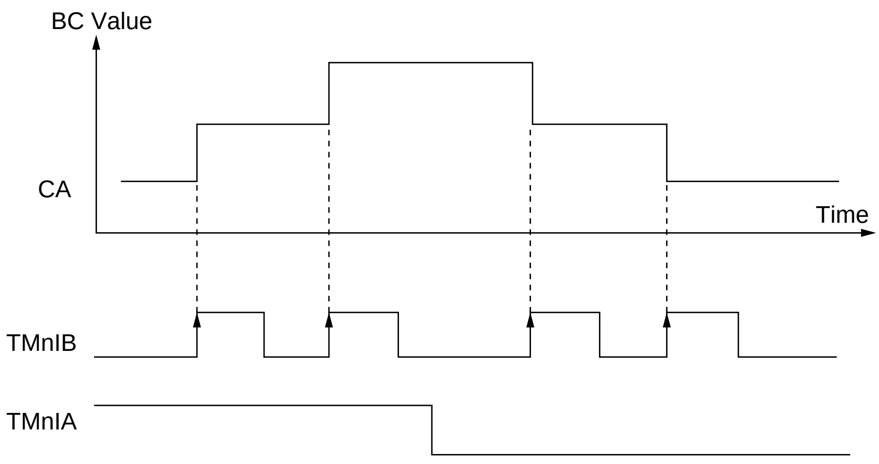


Figure 4-3-6 External Control Timing

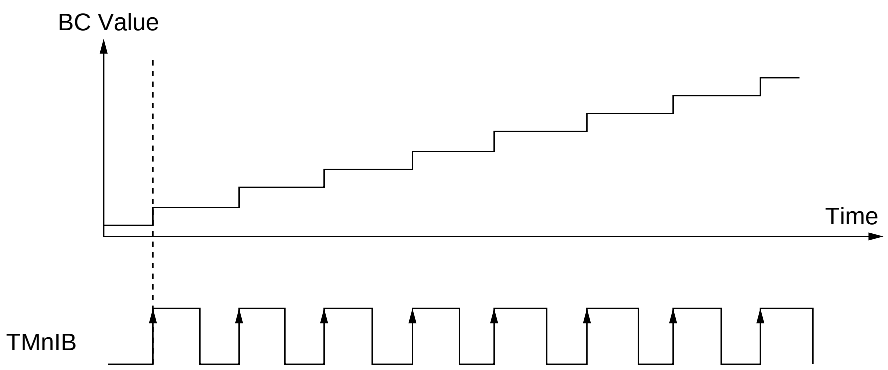


Figure 4-3-7 Event Counter Input Timing

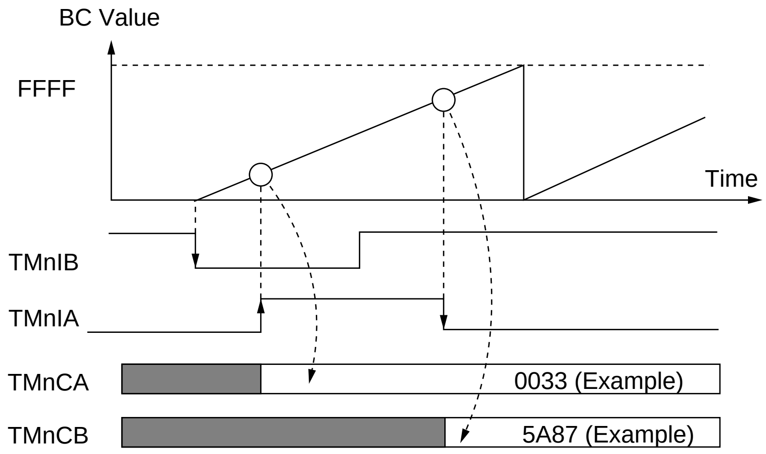


Figure 4-3-8 Input Capture 1 Timing

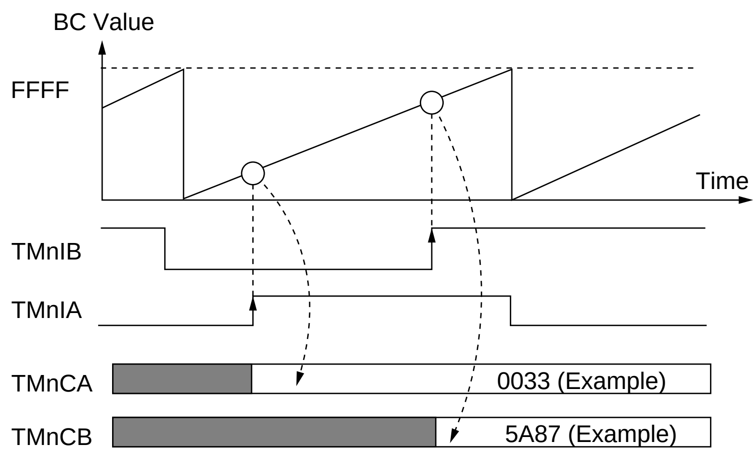


Figure 4-3-9 Input Capture 2 Timing

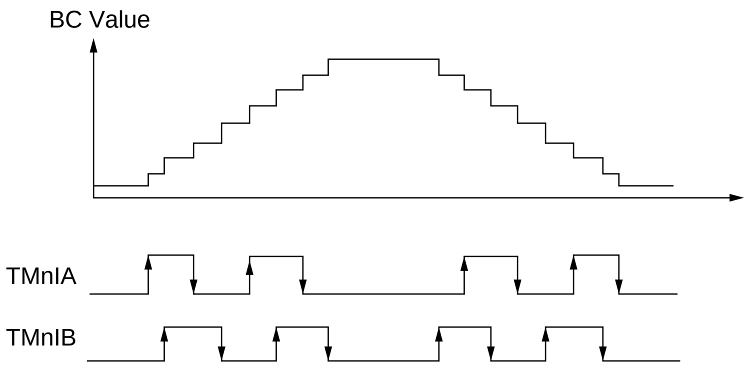


Figure 4-3-10 Two-phase Encoder (4x) Timing

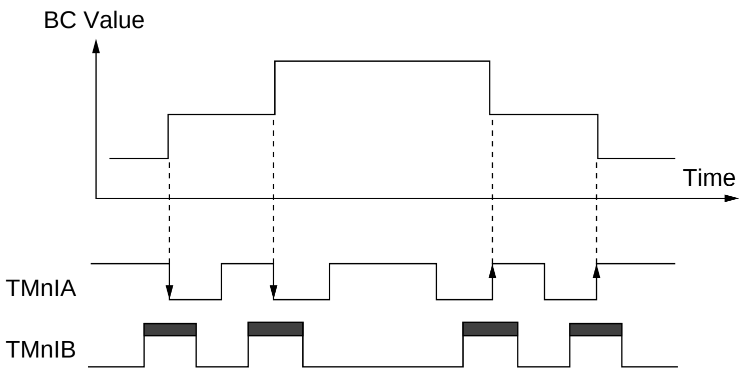


Figure 4-3-11 Two-phase Encoder (1x) Timing

4-3-2 16-bit Timer Control Registers

The timer binary counter (TMnBC), the timer compare/capture register A (TMnCA), the timer compare/capture register B (TMnCB) and the timer mode register (TMnMD) control 16-bit timer/counter functions.

TMn BC15	TMn BC14	TMn BC13	TMn BC12	TMn BC11	TMn BC10	TMn BC9	TMn BC8	TMn BC7	TMn BC6	TMn BC5	TMn BC4	TMn BC3	TMn BC2	TMn BC1	TMn BC0	**TMnBC [R]**

TMn CA15	TMn CA14	TMn CA13	TMn CA12	TMn CA11	TMn CA10	TMn CA9	TMn CA8	TMn CA7	TMn CA6	TMn CA5	TMn CA4	TMn CA3	TMn CA2	TMn CA1	TMn CA0	**TMnCA [R/W]**

TMn CB15	TMn CB14	TMn CB13	TMn CB12	TMn CB11	TMn CB10	TMn CB9	TMn CB8	TMn CB7	TMn CB6	TMn CB5	TMn CB4	TMn CB3	TMn CB2	TMn CB1	TMn CB0	**TMnCB [R/W]**

Use the MOV instruction to set TMnCA register and TMnCB register and only use 16-bit write operations.

The timer compare/capture register set AX (TMnCAX) and the timer compare/capture register set BX (TMnCBX) are valid only when double buffer mode is selected in the compare register. These registers prevent PWM losses. The value cannot be written directly in these registers by software. TMnCA value and TMnCB value can write to TMnCAX and TMnCBX respectively by writing dummy data to TMnCAX and TMnCBX. TMnCAX and TMnCBX registers cannot be read.

TMnMD (n: 8 to 12)

	TMnEN	TMnNLD	-	-	TMnUD1	TMnUD0	TMnTGE	TMnONE	TMnMD1	TMnMD0	TMnECLR	TMnLP	TMnASEL	TMnS2	TMnS1	TMnS0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

TMnS[2:0]	Clock Source Selection
000	Timer 0 underflow
001	Timer 4 underflow
010	TMnIOB
011	BOSC/2
100	Two-phase encoder (4x)
101	Two-phase encoder (1x)
110	TM8IC (only timer 8)
Other	Reserved

TMnASEL	TMnIOA Pin Output
0	RS.F.F output (One-phase PWM)
1	T.F.F output (Two-phase PWM)

TMnLP	TMnBC Count Range
0	0000 to FFFF
1	0000 to TMnCA value

TMnECLR	TMnBC Clear
0	Don't clear
1	Clear TMnBC when TMnIC=1 (Only timer 8 clear unconditionally)

TMnMD [1:0]	TMnCA, TMnCB Operating Mode Selection
00	Compare register (single buffer)
01	Compare register (double buffer)
10	Capture register (TMnIOA pin high or low)
11	Capture register (TMnIOA pin or TMnIOB pin high)

TMnONE	Counter Operating Mode Selection
0	Repeat (example: in PWM output mode)
1	One-shot counting (counting stops at the next cycle when TMnBC=TMnCA)

TMnTGE	Count Start External Trigger Enable
0	Disable
1	Start counting on the falling edge of TMnIOB

TMnUD [1:0]	Up/Down Operating Mode Selection
00	Up counter
01	Down counter
10	Up when TMnIOA is high; down when TMnIOA is low
11	Up when TMnIOB is high; down when TMnIOB is low

TMnNLD	TMnBC, T.F.F, and RS.F.F for Pin Output
0	Clear all (initialization)
1	Enable

TMnEN	TMnBC Count
0	Disable
1	Enable

Table 4-3-2 List of 16-bit Timer Control Registers

Register		Address	R/W	Function
Timer 8	TM8MD	x'00FE80'	R/W	Timer 8 Mode Register
	TM8BC	x'00FE82'	R	Timer 8 Binary Counter
	TM8CA	x'00FE84'	R/W	Timer 8 Compare/Capture Register A
	TM8CAX	x'00FE86'	-	Timer 8 Compare/Capture Register Set AX
	TM8CB	x'00FE88'	R/W	Timer 8 Compare/Capture Register B
	TM8CBX	x'00FE8A'	-	Timer 8 Compare/Capture Register Set BX
	TM8MD2	x'00FE8E'	R/W	Timer 8 Mode Register 2
Timer 9	TM9MD	x'00FE90'	R/W	Timer 9 Mode Register
	TM9BC	x'00FE92'	R	Timer 9 Binary Counter
	TM9CA	x'00FE94'	R/W	Timer 9 Compare/Capture Register A
	TM9CAX	x'00FE96'	-	Timer 9 Compare/Capture Register Set AX
	TM9CB	x'00FE98'	R/W	Timer 9 Compare/Capture Register B
	TM9CBX	x'00FE9A'	-	Timer 9 Compare/Capture Register Set BX
	TM9MD2	x'00FE9E'	R/W	Timer 9 Mode Register 2
Timer 10	TM10MD	x'00FEA0'	R/W	Timer 10 Mode Register
	TM10BC	x'00FEA2'	R	Timer 10 Binary Counter
	TM10CA	x'00FEA4'	R/W	Timer 10 Compare/Capture Register A
	TM10CAX	x'00FEA6'	-	Timer 10 Compare/Capture Register Set AX
	TM10CB	x'00FEA8'	R/W	Timer 10 Compare/Capture Register B
	TM10CBX	x'00FEAA'	-	Timer 10 Compare/Capture Register Set BX
	TM10MD2	x'00FEAE'	R/W	Timer 10 Mode Register 2
Timer 11	TM11MD	x'00FEB0'	R/W	Timer 11 Mode Register
	TM11BC	x'00FEB2'	R	Timer 11 Binary Counter
	TM11CA	x'00FEB4'	R/W	Timer 11 Compare/Capture Register A
	TM11CAX	x'00FEB6'	-	Timer 11 Compare/Capture Register Set AX
	TM11CB	x'00FEB8'	R/W	Timer 11 Compare/Capture Register B
	TM11CBX	x'00FEBA'	-	Timer 11 Compare/Capture Register Set BX
Timer 12	TM12MD	x'00FEC0'	R/W	Timer 12 Mode Register
	TM12BC	x'00FEC2'	R	Timer 12 Binary Counter
	TM12CA	x'00FEC4'	R/W	Timer 12 Compare/Capture Register A
	TM12CAX	x'00FEC6'	-	Timer 12 Compare/Capture Register Set AX
	TM12CB	x'00FEC8'	R/W	Timer 12 Compare/Capture Register B
	TM12CBX	x'00FECA"	-	Timer 12 Compare/Capture Register Set BX

4-3-3 16-bit Timer Block Diagrams

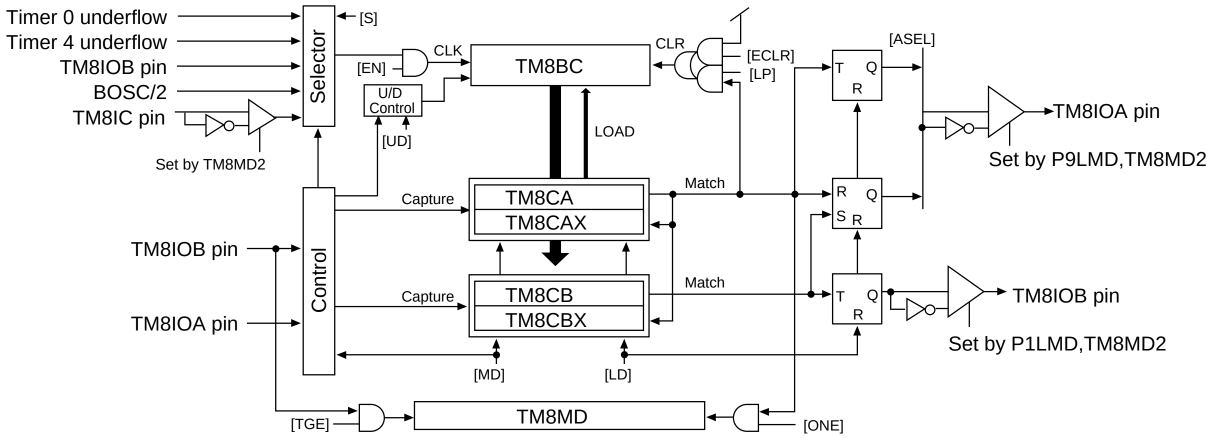


Figure 4-3-12 Timer 8 Block Diagram

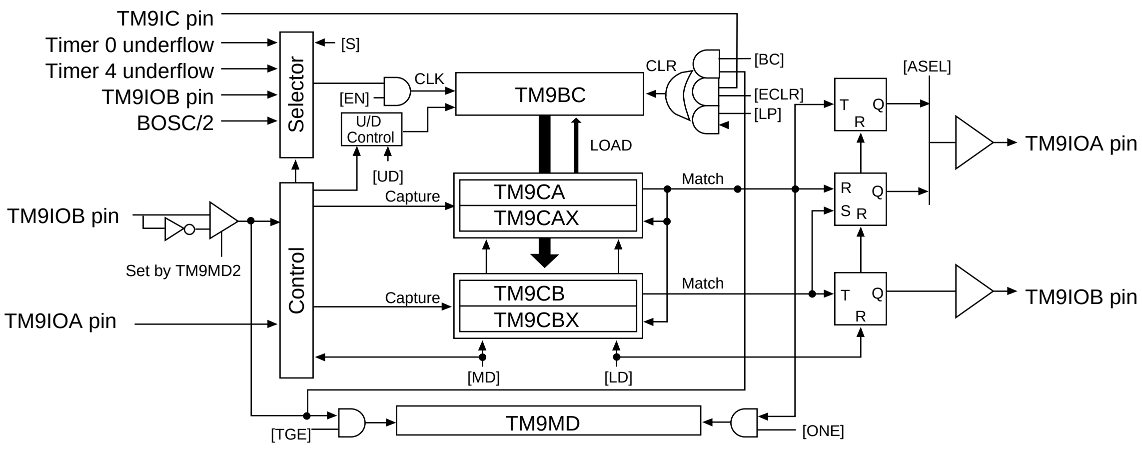


Figure 4-3-13 Timer 9 Block Diagram

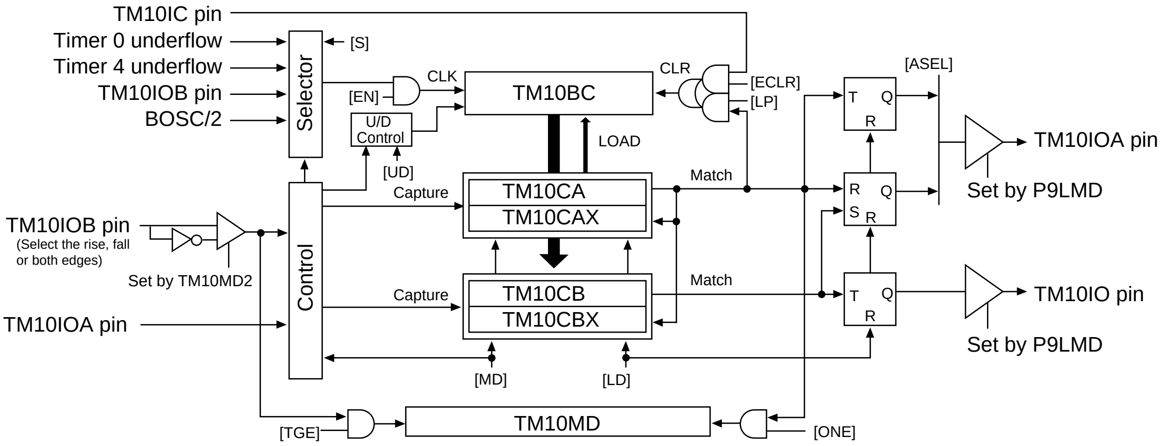


Figure 4-3-14 Timer 10 Block Diagram

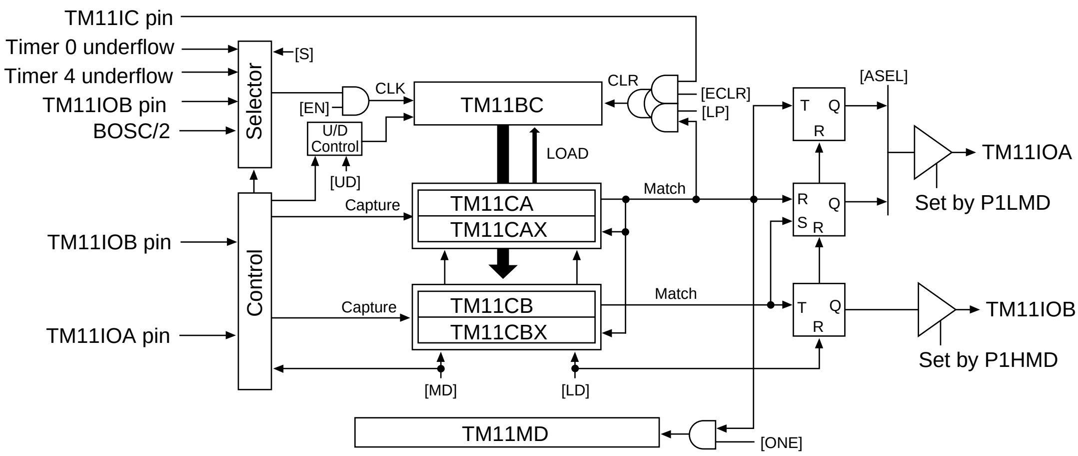


Figure 4-3-15 Timer 11 Block Diagram

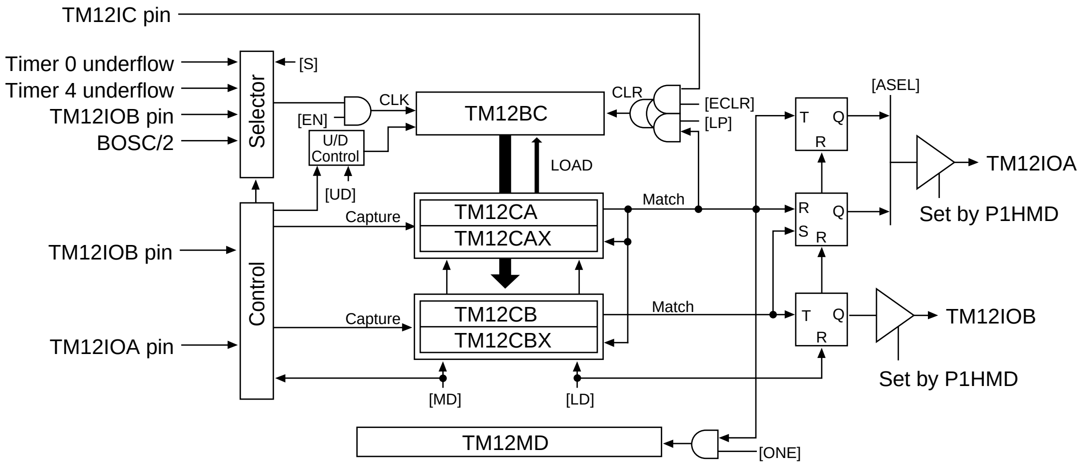
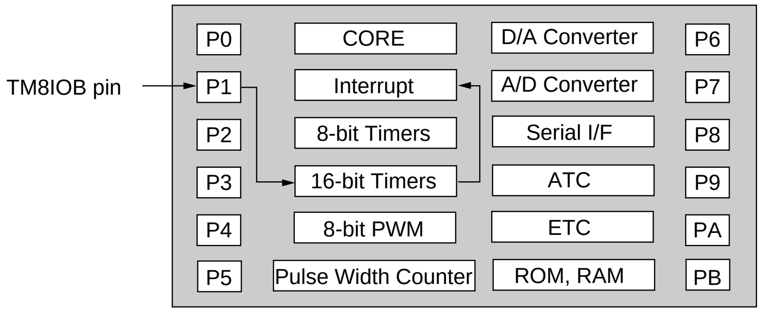


Figure 4-3-16 Timer 12 Block Diagram

4-4 16-bit Timer Setup Examples

4-4-1 Event Counter Using 16-bit Timer

In this example, timer 8 counts TM8IOB input (cycles of more than BOSC/4) and generates an interrupt on the second and fifth cycles.



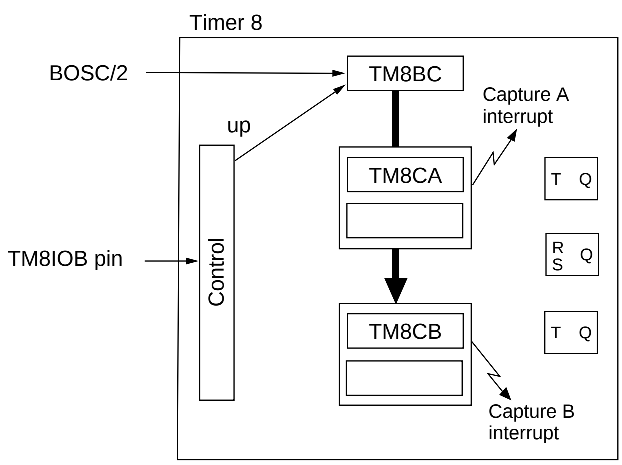


Figure 4-4-1 Event Counter Block Diagram

■ Timer 8 Setting

Use the MOV instruction to set the data and only use 16-bit write operations.

The timer 8 binary counter (TM8BC) is stopped, and TM8BC register and RS.F.F. are initialized (cleared to 0).

(1) Set the operating mode in the timer 8 mode register (TM8MD). Set counting stop and interrupt disable. Select up counting. Then, select TM8IOB pin as the clock source.

TM8MD: x'00FE80'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 EN	TM8 NLD			TM8 UD1	TM8 UD0	TM8 TGE	TM8 ONE	TM8 MD1	TM8 MD0	TM8 ECLR	TM8 LP	TM8 ASEL	TM8 S2	TM8 S1	TM8 S0
0	0			0	0	0	0	0	0	0	1	0	0	1	0

(2) Set the timer 8 divisor. Since timer 8 divides TM8IOB pin by 5, set the timer 8 compare/capture register A (TM8CA) to 4. (The valid range is 1 to x'FFFE'.)

TM8CA: x'00FE84'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 CA15	TM8 CA14	TM8 CA13	TM8 CA12	TM8 CA11	TM8 CA10	TM8 CA9	TM8 CA8	TM8 CA7	TM8 CA6	TM8 CA5	TM8 CA4	TM8 CA3	TM8 CA2	TM8 CA1	TM8 CA0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

(3) Set the phase difference for timer 8. Since the phase difference is 2 cycles of TM8IOB, set the timer 8 compare/capture register B (TM8CB) to 1. (The valid range is $0 \leq$ TM8CB < TM8CA.)

TM8CB: x'00FE88'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 CB15	TM8 CB14	TM8 CB13	TM8 CB12	TM8 CB11	TM8 CB10	TM8 CB9	TM8 CB8	TM8 CB7	TM8 CB6	TM8 CB5	TM8 CB4	TM8 CB3	TM8 CB2	TM8 CB1	TM8 CB0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

If this setting is omitted, the binary counter may not count the first cycle. Do not change to any other operating modes.

(4) Set TM8NLD and TM8EN of TM8MD register to 1 and 0 respectively. This enables TM8BC and RS.F.F.

(5) Set TM8NLD and TM8EN to 1. This starts the timer. Counting begins at the start of the next cycle.

■ **Interrupt Enable Setting**

(6) Enable interrupts after clearing all prior interrupt requests. To do this, set IQ0LV[2:0] of the external interrupt 0 control register (IQ0ICH) to the interrupt level 0 to 6), TM8AIR of the timer 8 capture A interrupt control register (TM8AICL) to 0, TM8BIR of the timer 8 capture B interrupt control register (TM8BICL) to 0, TM8AIE of TM8AICH register to 1 and TM8BIE of TM8BICH register to 1. Thereafter, a timer 8 capture A or B interrupt occurs when TM8BC counter matches TM8CA register or TM8CB register is generated.

Timer 8 functions as an event counter. Timer 8 does not operate stably when BOSC stops (in STOP mode). All external inputs are sampled on BOSC (synchronized with BOSC) when the external clock operates. The event counter frequency should be less than BOSC/4 (8.5 MHz with a 34-MHz oscillator).

Figure 4-4-2 shows an example of interrupt timing with an up counter.

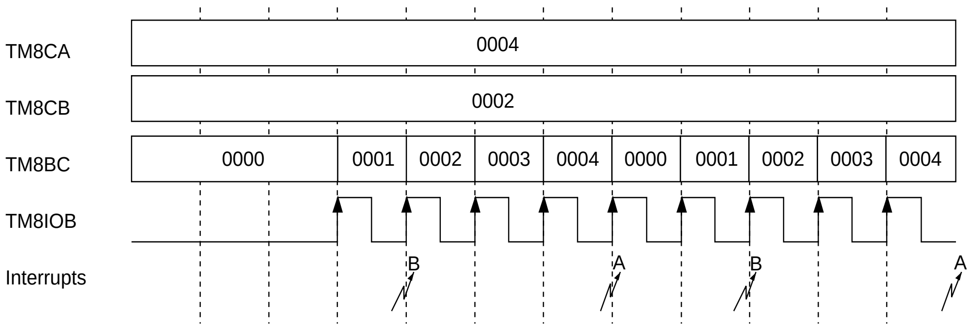


Figure 4-4-2 Event Counter Timing (16-bit Timer)

4-4-2 One-phase PWM Output Using 16-bit Timer

Timer 8 is used to divide BOSC/2 by 5 and outputs a one-phase PWM on the fifth cycle. The signal duty is 2:3. To do this, set the compare/capture register A to the divisor of 5 (set value is 4) and the compare/capture register B to the cycle of 2 (the set value is 1).

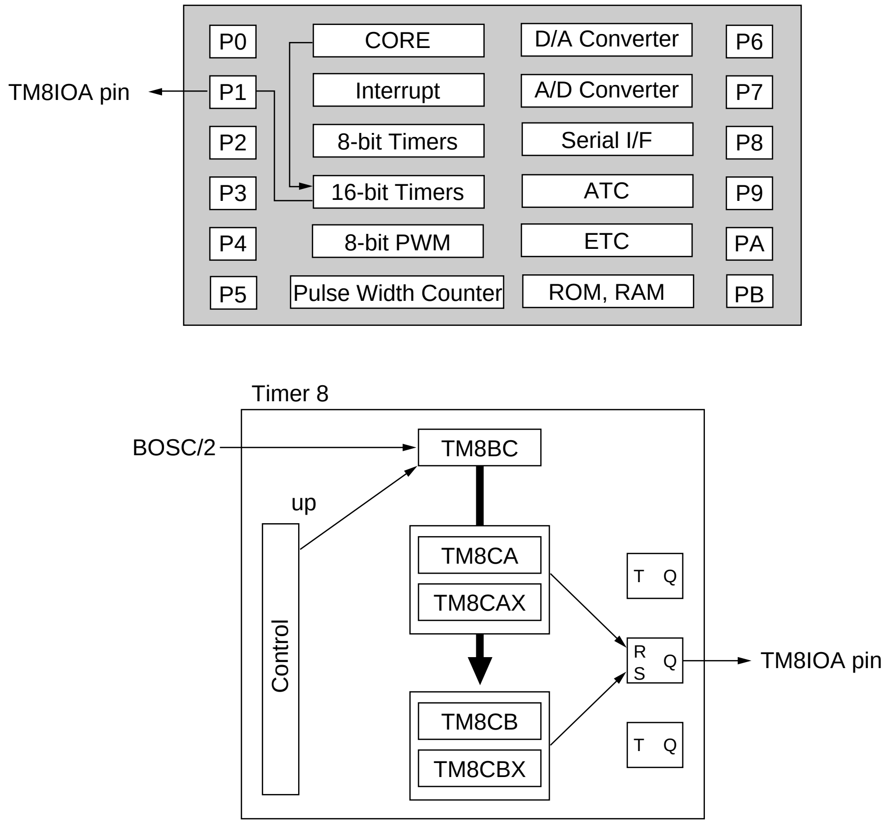


Figure 4-4-3 One-phase PWM Output Block Diagram (16-bit Timer)

■ Timer 8 Setting

(1) Set the operating mode in the timer 8 mode register (TM8MD). Set counting stop and interrupt disable. Select up counting. Select BOSC/2 as the clock source. Select the double buffer operating mode.

TM8MD: x'00FE80'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 EN	TM8 NLD			TM8 UD1	TM8 UD0	TM8 TGE	TM8 ONE	TM8 MD1	TM8 MD0	TM8 ECLR	TM8 LP	TM8 ASEL	TM8 S2	TM8 S1	TM8 S0
0	0			0	0	0	0	0	1	0	1	0	0	1	1

Use the MOV instruction to set the data and only use 16-bit write operations.

The timer 8 binary counter (TM8BC) is stopped, and TM8BC register and RS.F.F. are initialized (cleared to 0).

(2) Set the timer 8 divisor. Since the divisor is BOSC/2 divided by 5, set the timer 8 compare/capture register A (TM8CA) to 4. (The valid range is 1 to x'FFFE'.)

TM8CA: x'00FE84'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 CA15	TM8 CA14	TM8 CA13	TM8 CA12	TM8 CA11	TM8 CA10	TM8 CA9	TM8 CA8	TM8 CA7	TM8 CA6	TM8 CA5	TM8 CA4	TM8 CA3	TM8 CA2	TM8 CA1	TM8 CA0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

(3) Set the phase difference for timer 8. Since the phase difference is 2/5 cycles of BOSC/2, set the timer 8 compare/capture register B (TM8CB) to 1. (The valid range is $0 \leq$ TM8CB < TM8CA.)

TM8CB: x'00FE88'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 CB15	TM8 CB14	TM8 CB13	TM8 CB12	TM8 CB11	TM8 CB10	TM8 CB9	TM8 CB8	TM8 CB7	TM8 CB6	TM8 CB5	TM8 CB4	TM8 CB3	TM8 CB2	TM8 CB1	TM8 CB0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

(4) Write the dummy data to the timer 8 compare/capture register AX (TM8CAX) to set the initial value in the TM8CAX register. The value cannot be written directly in the TM8CAX register by software. In the double buffer mode, read the timer 8 compare/capture register A (TM8CA) to TM8CAX when TM8CAX=TM8BC. To read the TM8CA value into TM8CAX, write the dummy data to TM8CAX.

(5) Write the dummy data to the timer 8 compare/capture register BX (TM8CAX) to set the initial value in the TM8CBX register. The value cannot be written directly in the TM8CBX register by software. In the double buffer mode, read the timer 8 compare/capture register B (TM8CB) to TM8CBX when TM8CBX=TM8BC. To read the TM8CB value into TM8CBX, write the dummy data to TM8CBX.

(6) Set TM8NLD and TM8EN of TM8MD to 1 and 0 respectively. This enables TM8BC and RS.F.F.

If this setting is omitted, the binary counter may not count the first cycle. Do not change to any other operating modes.

(7) Set TM8NLD and TM8EN to 1. This starts the timer. Counting begins at the start of the next cycle.

Timer 8 outputs a one-phase PWM at any duty. Select up counting. Do not use timer 8 when BOSC stops (in STOP mode). All external inputs are sampled on BOSC (synchronized with BOSC) when the external clock operates.

Set the cycle (0 to x'FFFE') to the TM8CA register, and the duty to the TM8CB register. When TM8BC = TM8CB, reset RS.F.F and TM8BC at the start of the next cycle. The following shows the relationship between the waveforms of PWM output and the value of TM8CB register.

1) When TM8CA ≤ TM8CB ≤ x'FFFE': all output waveforms consist of 0.
2) When 0 ≤ TM8CB < TM8CA: under the condition that the PWM cycle is TM8CA+1 of the clock source, output 0 if TM8BC equals to the value between 0 and TM8BC, and output 1 if TM8BC equals to the value between TM8CB+1 and TM8CA.
3) When TM8CB=x'FFFF': all output waveforms consist of 1.

Figure 4-4-4 shows the TM8IOA pin output waveforms when TM8CA=4. A capture A interrupt or a capture B interrupt occurs. A capture B interrupt occurs only when TM8CB is set to 0 to TM8CA, and does not occur when TM8CB is set to any other values. (TM8BC and the value cannot be matched.) In Figure 4-4-4, CLRBC8 means an internal signal which clears TM8BC, S8 means an internal signal which sets RS.F.F. for TM8IOA pin output. R8 shows an internal signal which resets RS.F.F. for TM8IOA pin output.

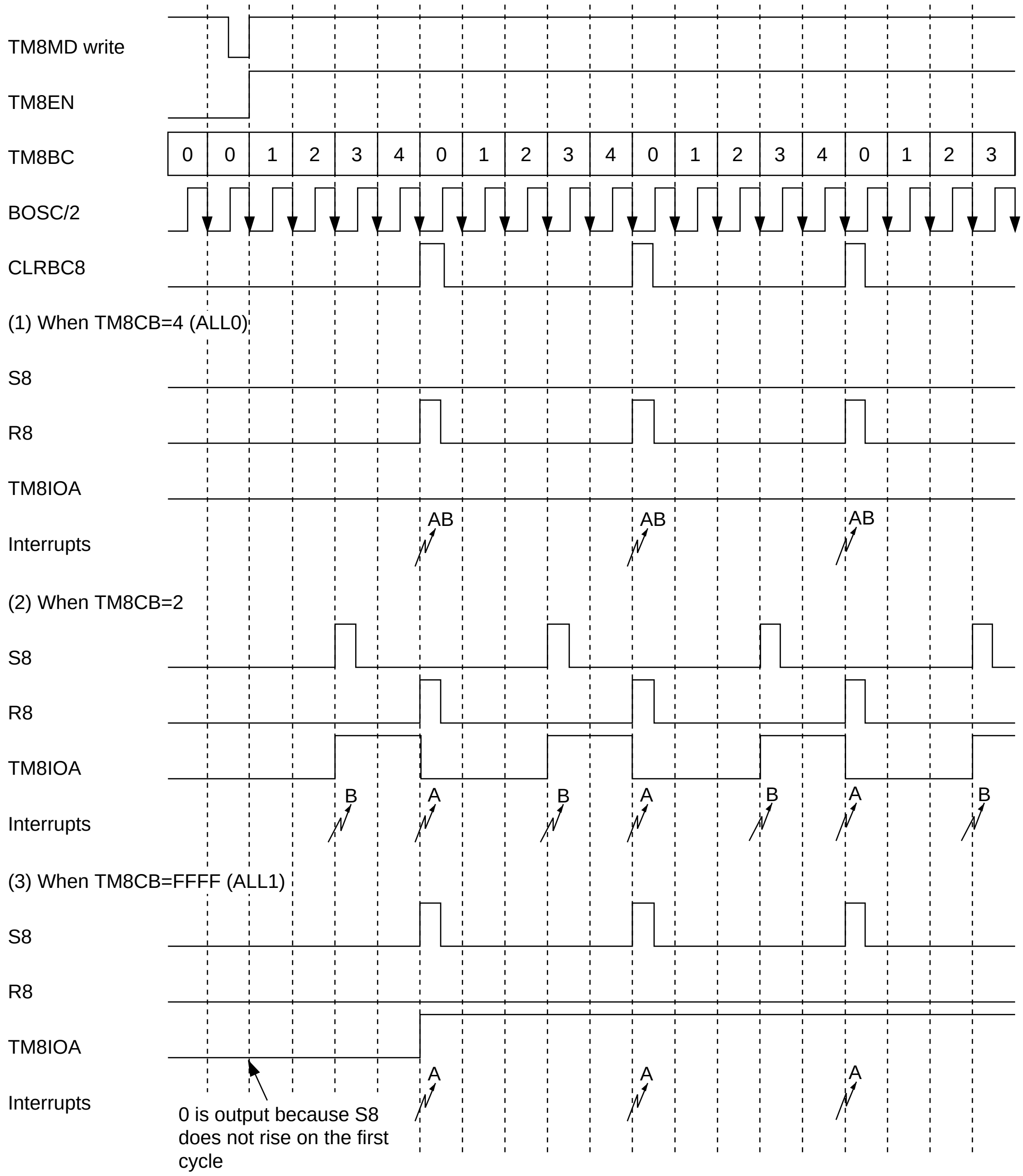


Figure 4-4-4 One-phase PWM Output Timing (16-bit Timer)

When outputting the PWM waveform, the timer may change the duty of the PWM output dynamically and may invert '1' and '0' due to the circuit configuration. This is caused when the trigger to be changed is lost based on the T.F.F output circuit. The RS.F.F. configuration in timer 8 prevents this error of inverting '1' and '0' at the trigger loss. In addition, the PWM waveform may be corrupted and interrupts are lost depending on the timing of changing the duty dynamically (in the single buffer mode on the figure below). In the double buffer mode, the duty can be changed from the next cycle, and the PWM loss does not occur at any timing of changing TMnCB. This loss does not occur even when the output waveforms consist of 1s or 0s.   Use double buffer mode normally when the PWM is used. Select single buffer mode based on applications.

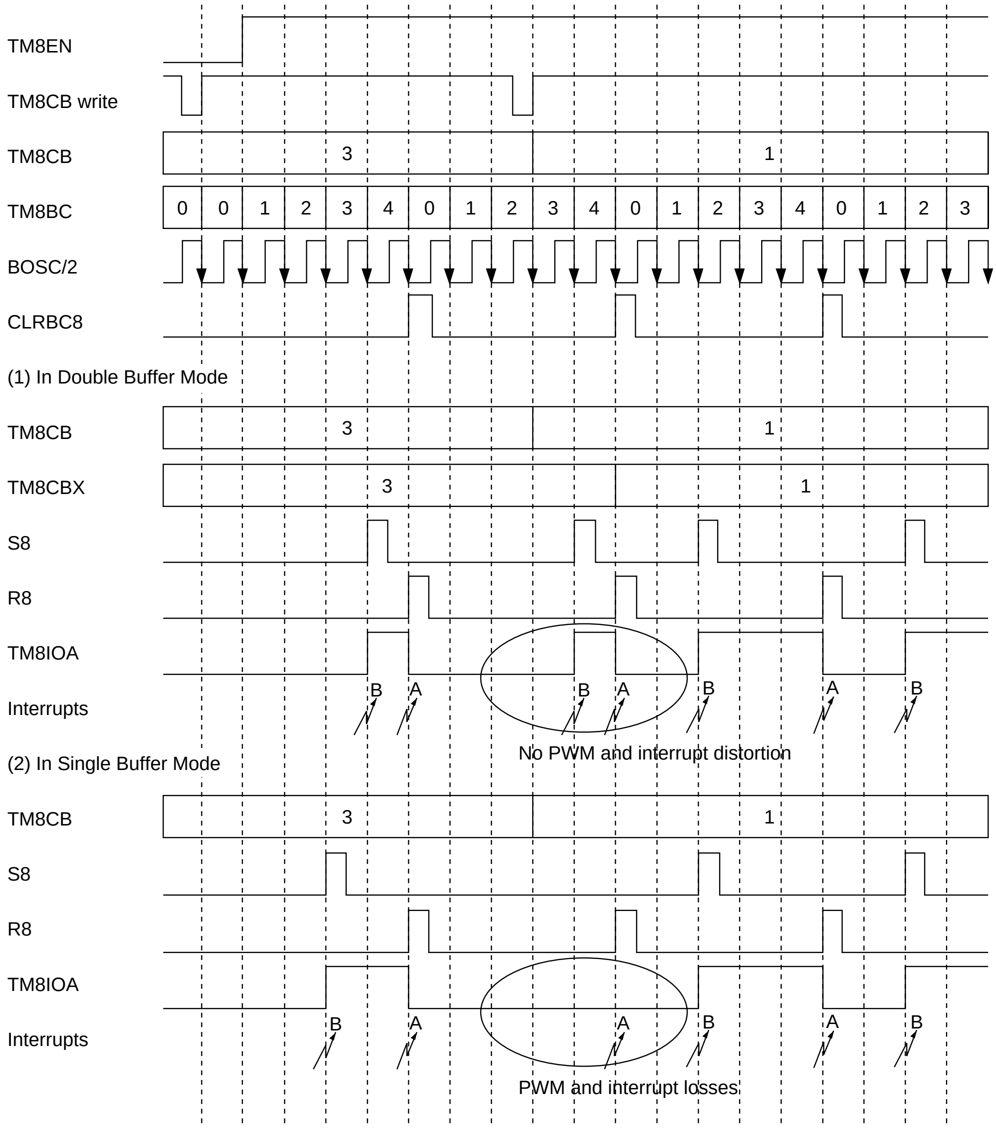


Figure 4-4-5 One-phase PWM Output Timing (16-bit Timer)
(Dynamical Duty Change)

4-4-3 Two-phase PWM Output Using 16-bit Timer

Timer 8 is used to divide timer 0 underflow by 5 and outputs a two-phase PWM on the fifth cycle. The phase difference is two cycles. To do this, set the timer 8 compare/capture register A to the divisor of 5 (set value is 4) and the timer 8 compare/capture register B to the cycle of 2 (the set value is 1).

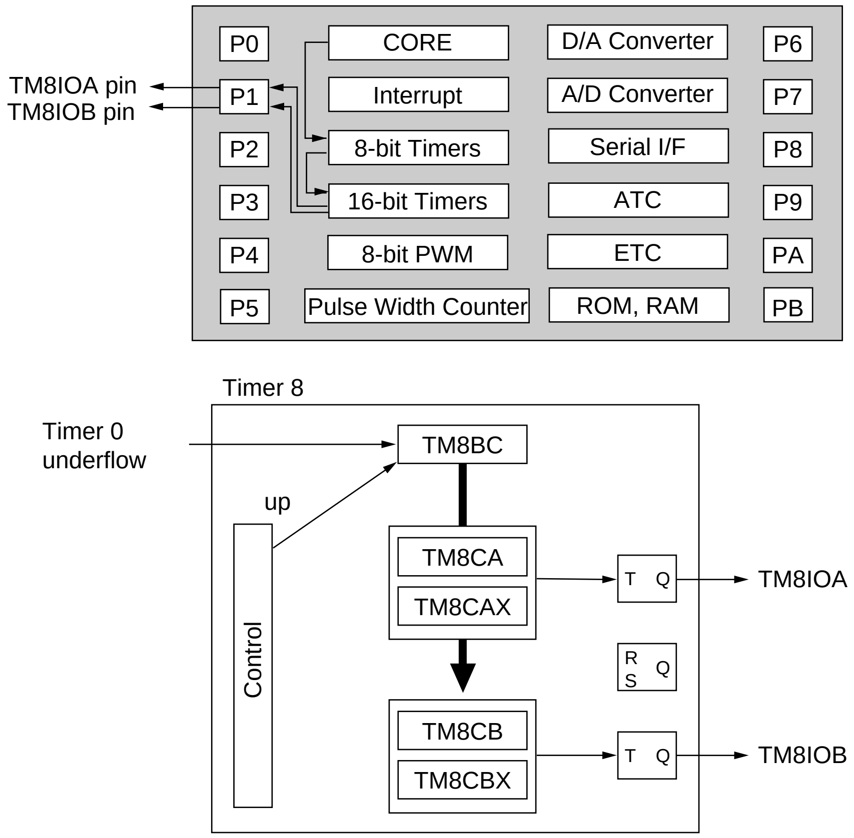


Figure 4-4-6 Two-phase PWM Output Block Diagram (16-bit Timer)

■ Timer 0 Setting

This verification is unnecessary after a reset.

(1) Verify that timer 0 counting is stopped with the timer 0 mode register (TM0MD).

TM0MD: x'00FE20'

7	6	5	4	3	2	1	0
TM0 EN	TM0 LD	-	-	-	-	TM0 S1	TM0 S0
0							

(2) Set the timer 0 divisor. Since timer 0 divides BOSC/2 by 2, set the timer 0 base register (TM0BR) to1. (The valid range for TM0BR is 0 to 255.)

TM0BR: x'00FE10'

7	6	5	4	3	2	1	0
TM0 BR7	TM0 BR6	TM0 BR5	TM0 BR4	TM0 BR3	TM0 BR2	TM0 BR1	TM0 BR0
0	0	0	0	0	0	0	1

Setting TM0EN and TM0LD to 0 is required between (3) and (4) in the bank address version and the linear address version, but this setting is not required in the linear address high-speed version.

(3) Load TM0BR value to the timer 0 binary counter (TM0BC). At the same time, select BOSC/2 as the clock source.

TM0MD: x'00FE20'

7	6	5	4	3	2	1	0
TM0 EN	TM0 LD	-	-	-	-	TM0 S1	TM0 S0
0	1					0	0

Do not change the clock source once you have selected it. Selecting the clock source while setting the count operation control will corrupt the value in the binary counter.

(4) Set TM0LD to 0 and TM0EN to 1. This starts the timer. Counting begins at the start of the next cycle.

When the timer 0 binary counter reaches 0 and the value 1 from the base register is loaded at the next count, a timer 0 underflow interrupt request will be sent to the CPU.

■ Timer 8 Setting

(1) Set the operating mode in the timer 8 mode register (TM8MD). Set counting stop. Select up counting. Select the timer 0 underflow as the clock source. Set T.F.F. as TM8IOA pin output. Select the double buffer mode in the compare register.

TM8MD: x'00FE80'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 EN	TM8 NLD			TM8 UD1	TM8 UD0	TM8 TGE	TM8 ONE	TM8 MD1	TM8 MD0	TM8 ECLR	TM8 LP	TM8 ASEL	TM8 S2	TM8 S1	TM8 S0
0	0			0	0	0	0	0	1	0	1	1	0	0	0

(2) Set the timer 8 divisor. Since the divisor is the timer 0 underflow divided by 5, set the timer 8 compare/capture register A (TM8CA) to 4. (The valid range is 1 to x'FFFE'.)

TM8CA: x'00FE84'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 CA15	TM8 CA14	TM8 CA13	TM8 CA12	TM8 CA11	TM8 CA10	TM8 CA9	TM8 CA8	TM8 CA7	TM8 CA6	TM8 CA5	TM8 CA4	TM8 CA3	TM8 CA2	TM8 CA1	TM8 CA0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

(3) Set the phase difference for timer 8. Since the phase difference is 2 cycles of the timer 0 underflow, set the timer 8 compare/capture register B (TM8CB) to 1. (The valid range is 0 ≤ TM8CB < TM8CA.)

TM8CB: x'00FE88'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 CB15	TM8 CB14	TM8 CB13	TM8 CB12	TM8 CB11	TM8 CB10	TM8 CB9	TM8 CB8	TM8 CB7	TM8 CB6	TM8 CB5	TM8 CB4	TM8 CB3	TM8 CB2	TM8 CB1	TM8 CB0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

(4) Write the dummy data to the timer 8 compare/capture register AX (TM8CAX) to set the initial value in the TM8CAX register. The value cannot be written in the TM8CAX by software. In the double buffer mode, read the timer 8 compare/capture register A (TM8CA) to TM8CAX when TM8CAX=TM8BC. To read the TM8CA value into TM8CAX, write the dummy data to TM8CAX.

Use the MOV instruction to set the data and only use 16-bit write operations.

The timer 8 binary counter (TM8BC) is stopped, and TM8BC register and RS.F.F. are initialized (cleared to 0).

(5) Write the dummy data to the timer 8 compare/capture register BX (TM8CAX) to set the initial value in the TM8CBX register. The value cannot be written in the TM8CBX by software. In the double buffer mode, read the timer 8 compare/capture register B (TM8CB) to TM8CBX when TM8CBX=TM8BC. To read the TM8CB value into TM8CBX, write the dummy data to TM8CBX.

If this setting is omitted, the binary counter may not count the first cycle. Do not change to any other operating modes.

(6) Set TM8NLD and TM8EN of TM8MD to 1 and 0 respectively. This enables TM8BC and RS.F.F.

(7) Set TM8NLD and TM8EN to 1. This starts the timer. Counting begins at the start of the next cycle.

Timer 8 outputs a one-phase PWM at any duty. Select up counting. Do not use timer 8 when BOSC stops (in STOP mode). All external inputs are sampled on BOSC (synchronized with BOSC) when the external clock operates.

Set the cycle (0 to x'FFFE') to the TM8CA register, and the phase difference to the TM8CB register. When TM8BC = TM8CB, generate a B8 pulse signal and invert T.F.F. for TM8IOB pin output at the start of the next cycle. When TM8BC = TM8CA, generate an A8 pulse signal, invert T.F.F. for TM8IOA pin output and reset TM8BC at the start of the next cycle. (A8 and B8 are internal control signals.)

Figure 4-4-7 shows the TM8IOA pin and TM8IOB pin output waveforms when TM8CA=4. A capture A interrupt and a capture B interrupt occur. Both interrupts occur at the start of the next cycle when TM8CA and TM8CB match. A capture B interrupt occurs only when TM8CB is set to 0 to TM8CA, and does not occur when TM8CB is set to any other value. (TM8BC and the value cannot be matched.)

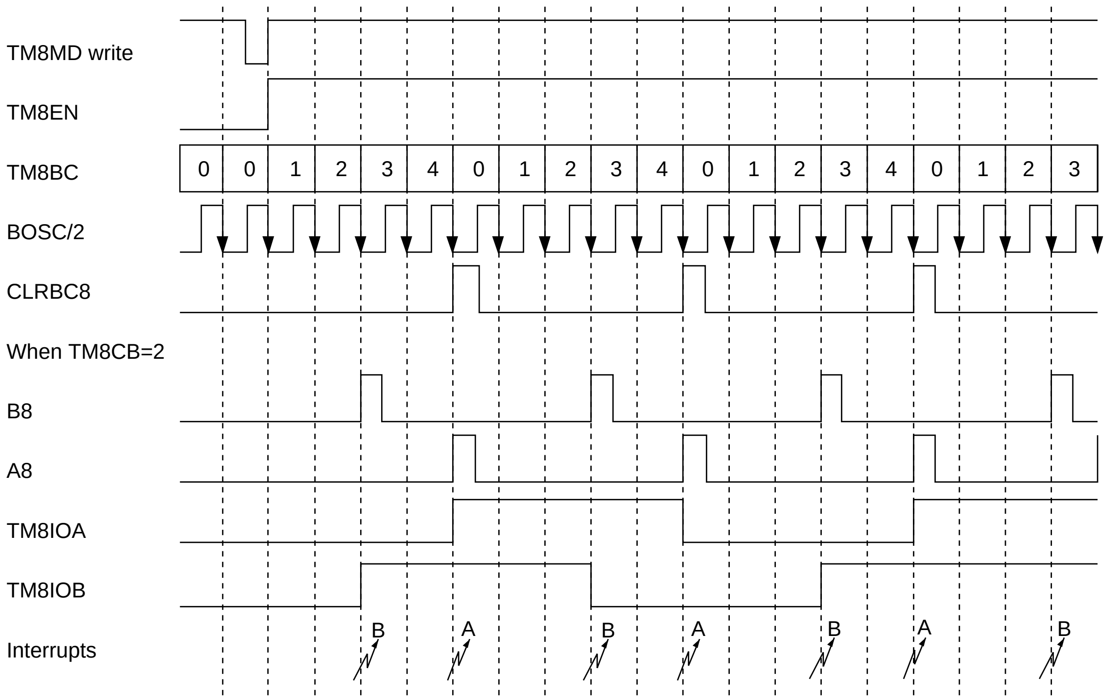


Figure 4-4-7 Two-phase PWM Output Timing (16-bit Timer)

When outputting the PWM waveform, the timer may change the duty of the PWM output dynamically. The PWM waveform may be corrupted and interrupts are lost depending on the timing of changing the duty dynamically (in the single buffer mode on the figure below). In the double buffer mode, the duty can be changed from the next cycle, and the PWM loss does not occur at any timing of changing TM8CB. This loss does not occur even when the output waveforms consist of 1s or 0s. Use double buffer mode when the PWM is used. Select single buffer mode depending on applications.

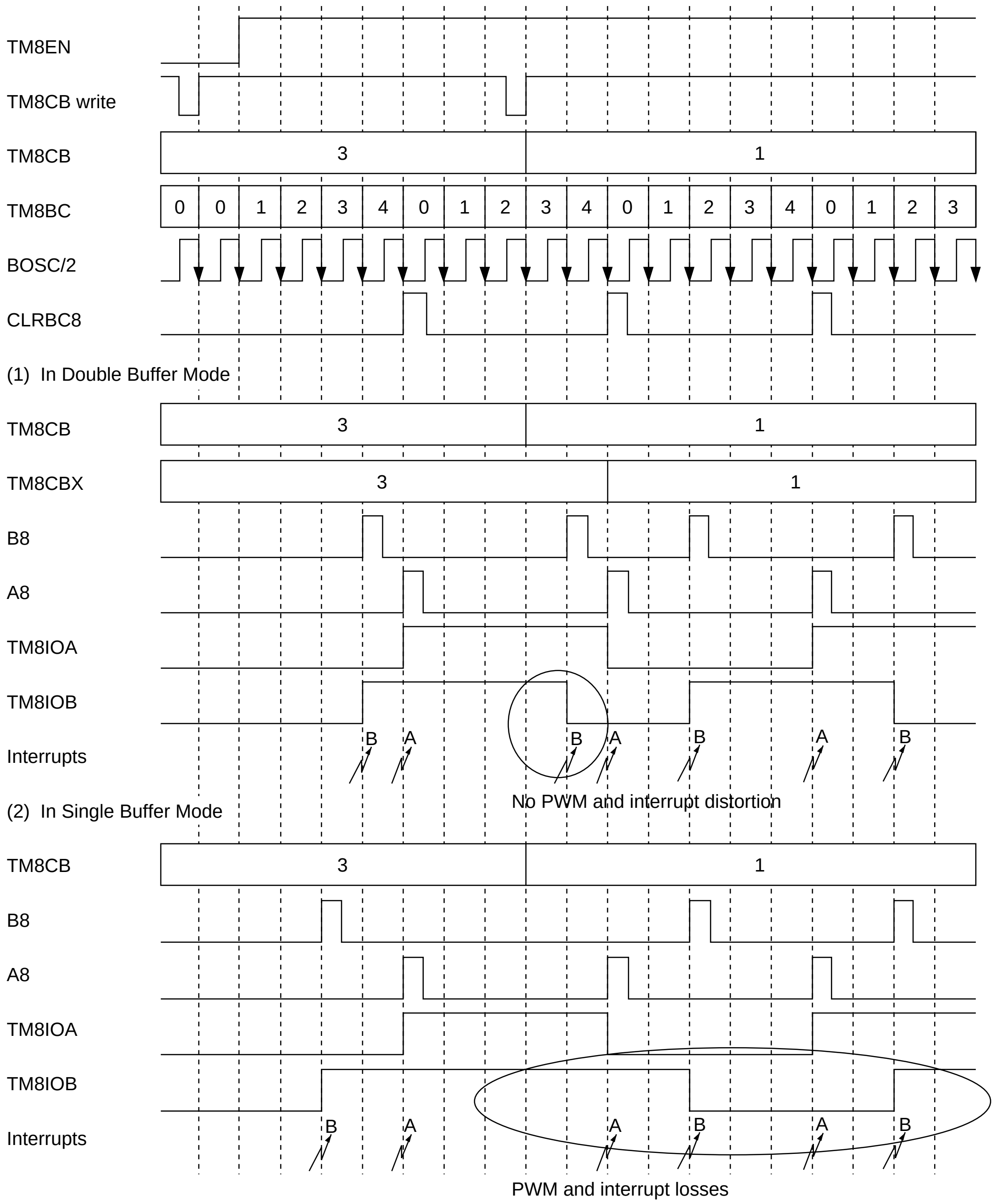


Figure 4-4-8 Two-phase PWM Output Timing (16-bit Timer)
(Dynamical Duty Change)

4-4-4 One-phase Capture Input Using 16-bit Timer

Timer 8 is used to divide BOSC/2 by 65,536 and measure how long TM8IOA input is high. An interrupt occurs on capture B. The width is calculated by the instruction (TM8CB-TM8CA).

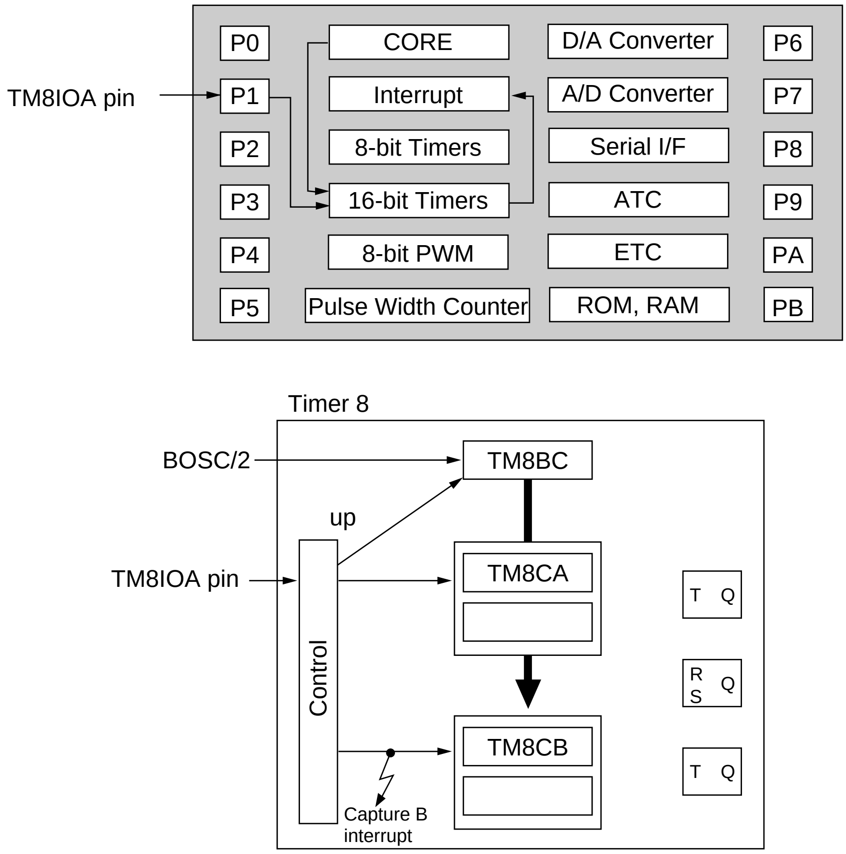


Figure 4-4-9 One-phase Capture Input Block Diagram (16-bit Timer)

■ Timer 8 Setting

Use the MOV instruction to set the data and only use 16-bit write operations.

The timer 8 binary counter (TM8BC) is stopped, and TM8BC register and RS.F.F. are initialized (cleared to 0).

(1) Set the operating mode in the timer 8 mode register (TM8MD). Set counting stop. Select up counting. Select BOSC/2 as the clock source. Set the operating mode of the capture register to capture on both edges of TM8IOA pin.

TM8MD: x'00FE80'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 EN	TM8 NLD			TM8 UD1	TM8 UD0	TM8 TGE	TM8 ONE	TM8 MD1	TM8 MD0	TM8 ECLR	TM8 LP	TM8 ASEL	TM8 S2	TM8 S1	TM8 S0
0	0			0	0	0	0	1	0	0	0	1or0	0	1	1

If this setting is omitted, the binary counter may not count the first cycle. Do not change to any other operating modes.

(2) Set TM8NLD and TM8EN of TM8MD to 1 and 0 respectively. This enables TM8BC and RS.F.F.

(3) Set TM8NLD and TM8EN to 1. This starts the timer. Counting begins at the start of the next cycle.

■ Interrupt Enable Setting

When TM8MD0=0 and TM8MD1=1 (in capture mode), TM8CA and TM8CB become read-only registers. If TM8CB must be set, TM8MD0 and TM8MD1 must be set to 0.

(4) Enable interrupts after clearing all prior interrupt requests. To do this, set IQ0LV[2:0] of the external interrupt 0 control register (IQ0ICH) to the interrupt level 0 to 6), TM8BIR of the timer 8 capture B interrupt control register (TM8BICL) to 0, and TM8BIE of the timer 8 capture B interrupt control register (TM8BICH) to 1. Thereafter, a timer 8 capture B interrupt occurs when the capture to the TM8CB register is generated on the rising edge of TM8IOA pin.

■ Interrupt Processing and Signal Width Calculation

(5) Execute the interrupt service routine. The interrupt service routine determines the interrupt group and factor, and clears TM8BIR flag.

(6) Calculate the signal width. Save the TM8CA value and the TM8CB value to the data registers (DR0 to DR3), and subtract the TM8CA value from the TM8CB value. The width will be calculated correctly even if the TM8CA value is greater than the TM8CB value by setting TM8LP to 0. Figure 4-4-10 shows 000A-0007=0003, or 3 cycles.

Timer 8 functions as a one-phase capture input. Select up counting. Timer 8 does not operate stably when BOSC stops (in STOP mode). All external inputs are sampled on BOSC (synchronized with BOSC) when the external clock operates. TM8CA is captured on the rising edge of TM8IOA, and TM8CB is captured on the falling edge of TM8IOA. A capture B interrupt is generated on the TM8CB capture, and the TM8CA and TM8CB values are read during the interrupt service routine. Figure 4-4-10 shows 000A-0007=0003, or 3 cycles. The same result is obtained even if the TM8CA value is greater than the TM8CB value. For example, 0003-FFFE=0005. The signal width is calculated by ignoring flags.

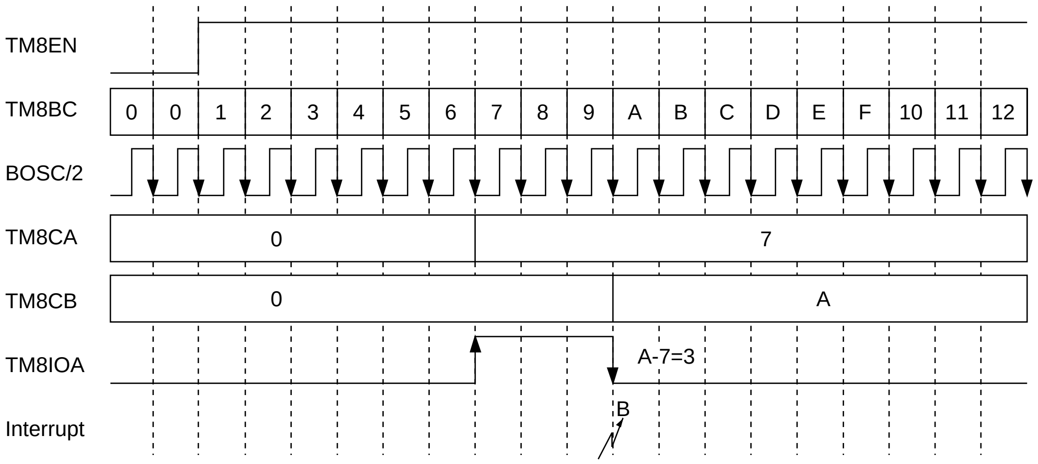


Figure 4-4-10 One-phase Capture Input Timing (16-bit Timer)

4-4-5 Two-phase Capture Input Using 16-bit Timer

Timer 8 is used to divide timer 0 underflow by 65,536 and measures the width from the rising edge of the TM8IOA input to the rising edge of TM8IOB input. An interrupt occurs on capture B. The width is calculated by the instruction (TM8CB-TM8CA).

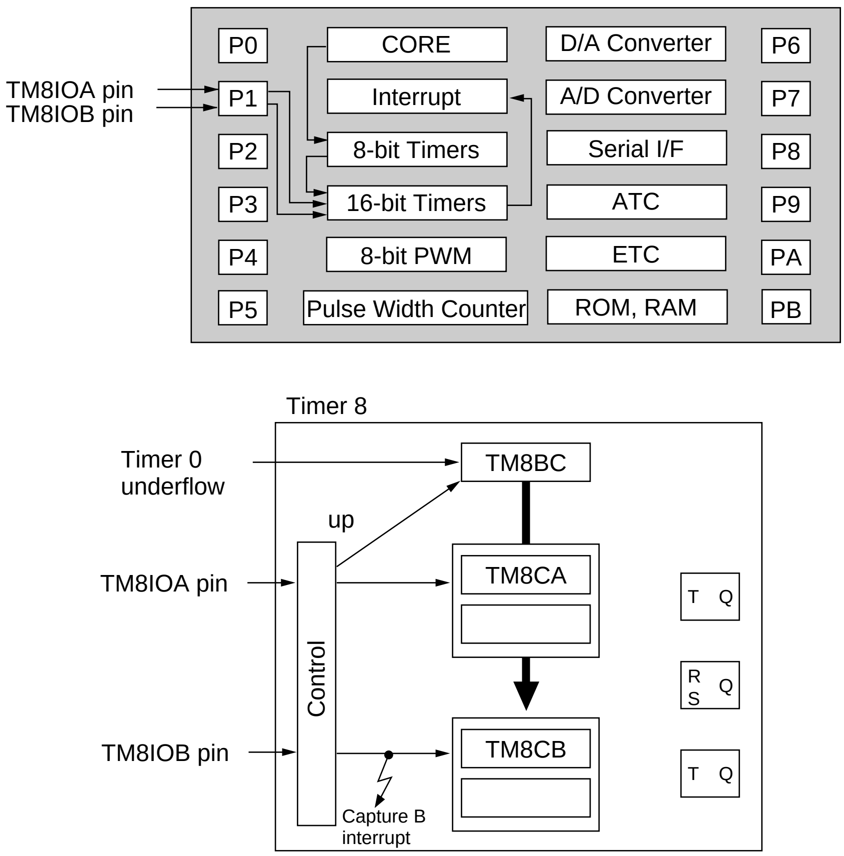


Figure 4-4-11 Two-phase Capture Input Block Diagram (16-bit Timer)

■ Timer 0 Setting

(1) Verify that timer 0 counting is stopped with the timer 0 mode register (TM0MD).

This verification is unnecessary after a reset.

TM0MD: x'00FE20'

7	6	5	4	3	2	1	0
TM0 EN	TM0 LD	-	-	-	-	TM0 S1	TM0 S0
0							

(2) Set the timer 0 divisor. In this example, since timer 0 divides BOSC/2 by 2, set the timer 0 base register (TM0BR) to1. (The valid range for TM0BR is 0 to 255.)

TM0BR: x'00FE10'

7	6	5	4	3	2	1	0
TM0 BR7	TM0 BR6	TM0 BR5	TM0 BR4	TM0 BR3	TM0 BR2	TM0 BR1	TM0 BR0
0	0	0	0	0	0	0	1

(3) Load TM0BR value to the timer 0 binary counter (TM0BC). At the same time, select BOSC/2 as the clock source.

Setting TM0EN and TM0LD to 0 is required between (3) and (4) in the bank address version and the linear address version, but this setting is not required in the linear address high-speed version.

TM0MD: x'00FE20'

7	6	5	4	3	2	1	0
TM0 EN	TM0 LD	-	-	-	-	TM0 S1	TM0 S0
0	1					0	0

(4) Set TM0LD to 0 and TM0EN to 1. This starts the timer. Counting begins at the start of the next cycle.

Do not change the clock source once you have selected it. Selecting the clock source while setting the count operation control will corrupt the value in the binary counter.

When TM0BC reaches 0 and the value 1 from the TM0BR register is loaded at the next count, a timer 0 underflow interrupt request will be sent to the CPU.

Use the MOV instruction to set the data and only use 16-bit write operations.

The timer 8 binary counter (TM8BC) is stopped, and TM8BC register and RS.F.F. are initialized (cleared to 0).

If this setting is omitted, the binary counter may not count the first cycle. Do not change to any other operating modes.

When TM8MD0=0 and TM8MD1=1 (in capture mode), TM8CA and TM8CB become read-only registers. If TM8CB must be set, TM8MD0 and TM8MD1 must be set to 0.

■ Timer 8 Setting

(1) Set the operating mode in the timer 8 mode register (TM8MD). Set counting stop. Select up counting. By setting TM8NLD of the TM8MD register to 1, select repeat counting from 0 to x'FFFF'. Select timer 0 underflow as the clock source. Set the operating mode of the capture register to the rising edge of TM8IOA pin and the rising edge of TM8IOB pin.

TM8MD: x'00FE80'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 EN	TM8 NLD			TM8 UD1	TM8 UD0	TM8 TGE	TM8 ONE	TM8 MD1	TM8 MD0	TM8 ECLR	TM8 LP	TM8 ASEL	TM8 S2	TM8 S1	TM8 S0
0	0			0	0	0	0	1	1	0	0	1or0	0	0	0

(2) Set TM8NLD and TM8EN of TM8MD to 1 and 0 respectively. This enables TM8BC and RS.F.F.

(3) Set TM8NLD and TM8EN to 1. This starts the timer. Counting begins at the start of the next cycle.

■ Interrupt Enable Setting

(4) Enable interrupts after clearing all prior interrupt requests. To do this, set IQ0LV[2:0] of the external interrupt 0 control register (IQ0ICH) to the interrupt level 0 to 6), TM8BIR of the timer 8 capture B interrupt control register (TM8BICL) to 0, and TM8BIE of the timer 8 capture B interrupt control register (TM8BICH) to 1. Thereafter, a timer 8 capture B interrupt occurs when the capture to TM8CB register is generated on the rising edge of TM8IOB pin.

■ Interrupt Processing and Signal Width Calculation

(5) Execute the interrupt service routine. The interrupt service routine determines the interrupt group and factor, and clears TM8BIR flag.

(6) Calculate the signal width. Save the TM8CA value and the TM8CB value to the data registers (DR0 to DR3), and subtract the TM8CA value from the TM8CB value. The width will be calculated correctly even if the TM8CA value is greater than the TM8CB value by setting TM8LP to 0. Figure 4-4-12 shows 000A-0007=0003, or 3 cycles.

Timer 8 functions as a one-phase capture input. Select up counting. Timer 8 does not operate stably when BOSC stops (in STOP mode). All external inputs are sampled on BOSC (synchronized with BOSC) when the external clock operates. The TM8CA register is captured on the rising edge of TM8IOA pin, and the TM8CB register is captured on the rising edge of TM8IOB pin. A capture B interrupt is generated on the TM8CB capture, and the TM8CA and TM8CB values are read during the interrupt processing routine.   Figure 4-4-12 shows 000A-0007=0003, or 3 cycles. The same result is obtained even if the TM8CA value is greater than the TM8CB value. For example, 0003-FFFE=0005. The signal width is calculated by ignoring flags.

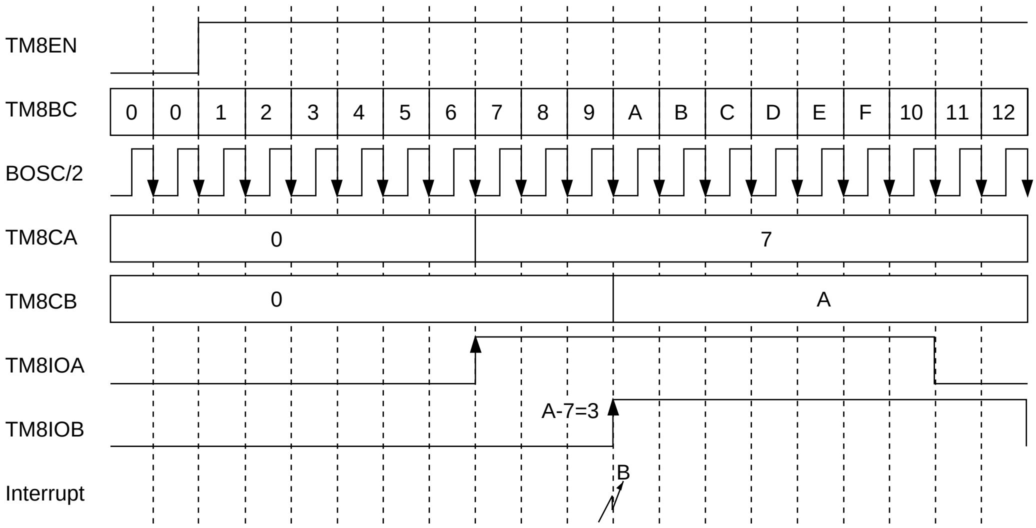


Figure 4-4-12 Two-phase Capture Input Timing (16-bit Timer)

4-4-6 Two-phase Encoder Input (4x) Using 16-bit Timer

Timer 8 receives a two-phase encoder input (4x) and counts up and down. An interrupt occurs when the TM8BC value reached the TM8CA value or the TM8CB value.

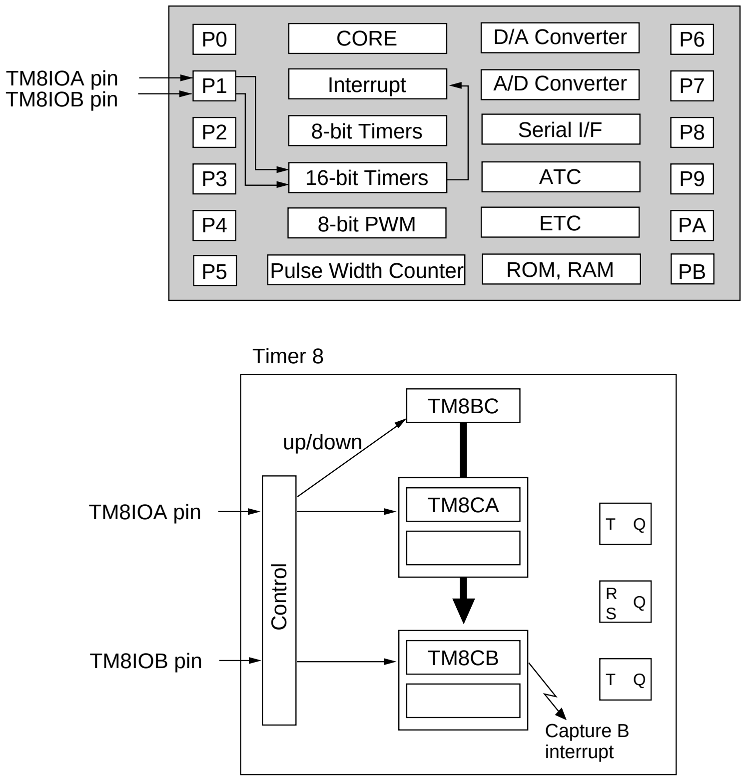


Figure 4-4-13 Two-phase Encoder Input (4x) Block Diagram (16-bit Timer)

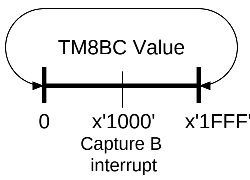


Figure 4-4-14 Two-phase Encoder Input (4x) Configuration Example 1

As Figure 4-4-15 shown, it is possible to set capture A interupt and capture B interrupt in different places separately. (Setting TM8LP of the TM8MD register to 0 is required.)

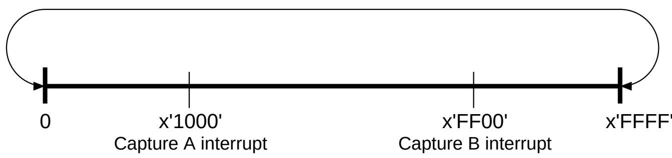


Figure 4-4-15 Two-phase Encoder Input (4x) Configuration Example 2

■ Timer 8 Setting

(1) Set the operating mode in the timer 8 mode register (TM8MD). Set counting stop. Count setting is ignored. Since counting is performed by looping on the TM8CA value, set TM8LP of the TM8MD register to 1. Select the two-phase encoder (4x) as the clock source.

Use the MOV instruction to set the data and only use 16-bit write operations.

The timer 8 binary counter (TM8BC) is stopped, and TM8BC register and RS.F.F. are initialized (cleared to 0).

TM8MD: x'00FE80'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 EN	TM8 NLD			TM8 UD1	TM8 UD0	TM8 TGE	TM8 ONE	TM8 MD1	TM8 MD0	TM8 ECLR	TM8 LP	TM8 ASEL	TM8 S2	TM8 S1	TM8 S0
0	0			0	0	0	0	0	0	0	1	1or0	1	0	0

(2) Set the timer 8 looping value to the TM8CA register (the valid range: 1 to x'FFFF'). The TM8BC counter counts from 0 to x'1FFF' when writing x'1FFF' to the TM8CA register.

TM8CA: x'00FE84'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 CA15	TM8 CA14	TM8 CA13	TM8 CA12	TM8 CA11	TM8 CA10	TM8 CA9	TM8 CA8	TM8 CA7	TM8 CA6	TM8 CA5	TM8 CA4	TM8 CA3	TM8 CA2	TM8 CA1	TM8 CA0
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1

(3) Set the timer 8 interrupt value to the TM8CB register (the valid range: 0 to TM8CA). In this example, write x'1000'. Whenever the up or down counter reaches this value, a capture B interrupt occurs at the beginning of the next cycle.

TM8CB: x'00FE88'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 CB15	TM8 CB14	TM8 CB13	TM8 CB12	TM8 CB11	TM8 CB10	TM8 CB9	TM8 CB8	TM8 CB7	TM8 CB6	TM8 CB5	TM8 CB4	TM8 CB3	TM8 CB2	TM8 CB1	TM8 CB0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

(4) Set TM8NLD and TM8EN of the TM8MD register to 1 and 0 respectively. This enables TM8BC and RS.F.F.

If this setting is omitted, the binary counter may not count the first cycle. Do not change to any other operating modes.

(5) Set TM8NLD and TM8EN of the TM8MD register to 1. This starts the timer. Counting begins at the start of the next cycle.

■ **Interrupt Enable Setting**

(6) Enable interrupts after clearing all prior interrupt requests. To do this, set IQ0LV[2:0] of the external interrupt 0 control register (IQ0ICH) to the interrupt level 0 to 6), TM8BIR of the timer 8 capture B interrupt control register (TM8BICL) to 0, and TM8BIE of the timer 8 capture B interrupt control register (TM8BICH) to 1. Thereafter, a timer 8 capture B interrupt occurs when the TM8BC counter matches TM8CB register.

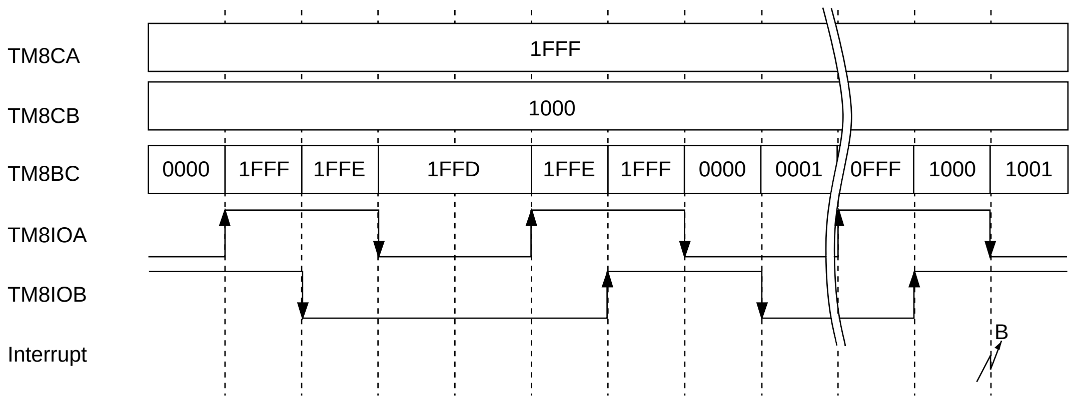
■ **Interrupt Processing**

(7) First, determine the interrupt group and factor, and clear TM8BIR flag during the interrupt service routine.

(8) Execute the interrupt service routine.

Timer 8 functions as a two-phase encoder input. Timer 8 does not operate stably when BOSC stops (in STOP mode). All external inputs are sampled on BOSC (synchronized with BOSC) when the external clock operates.

Figure 4-4-16 shows the counting direction. When counting down, the next value after 0 becomes the TM8CA value. When the TM8BC value matches the TM8CB value, a capture B interrupt occurs.

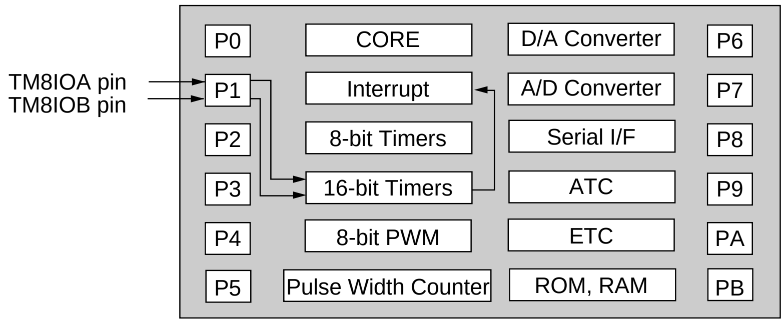


	Up Counting	Down Counting
TM8IOA	↑ 1 ↓ 0	↑ 0 ↓ 1
TM8IOB	0 ↑ 1 ↓	1 ↑ 0 ↓

Figure 4-4-16 Two-phase Encoder Input Timing (4x) (16-bit Timer)

4-4-7 Two-phase Encoder Input (1x) Using 16-bit Timer

Timer 8 receives a two-phase encoder input (4x) and counts up and down. An interrupt occurs when the preset value is reached.



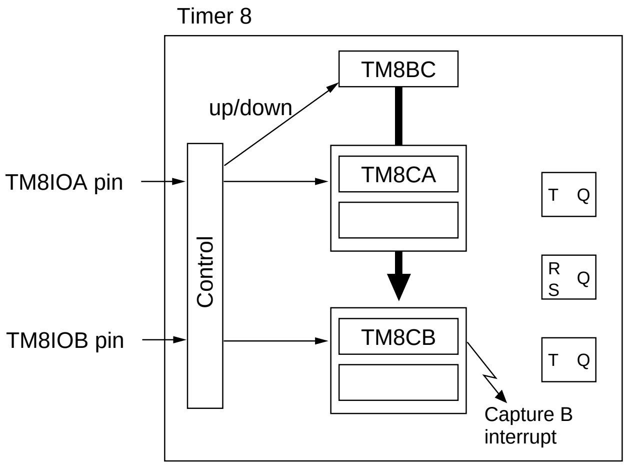


Figure 4-4-17 Two-phase Encoder Input (1x) Block Diagram (16-bit Timer)

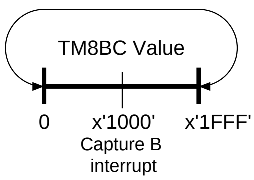


Figure 4-4-18 Two-phase Encoder Input (1x) Configuration Example 1

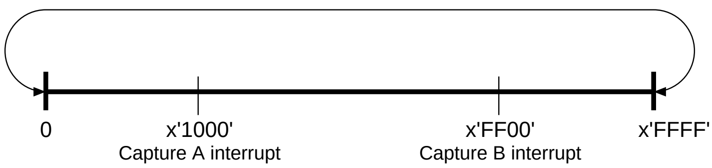


Figure 4-4-19 Two-phase Encoder Input (1x) Configuration Example 2

As Figure 4-4-19 shown, it is possible to set capture A interrupt and capture B interrput in different places separately. (Setting TM8LP of the TM8MD register to 0 is required.)

Use the MOV instruction to set the data and only use 16-bit write operations.

■ Timer 8 Setting

(1) Set the operating mode in the timer 8 mode register (TM8MD). Set counting stop. Count setting is ignored. Since counting is performed by looping on the TM8CA value, set TM8LP of the TM8MD register to 1. Select the two-phase encoder (1x) as the clock source.

The timer 8 binary counter (TM8BC) is stopped, and TM8BC register and RS.F.F. are initialized (cleared to 0).

TM8MD: x'00FE80'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 EN	TM8 NLD			TM8 UD1	TM8 UD0	TM8 TGE	TM8 ONE	TM8 MD1	TM8 MD0	TM8 ECLR	TM8 LP	TM8 ASEL	TM8 S2	TM8 S1	TM8 S0
0	0			0	0	0	0	0	0	0	1	1or0	1	0	1

(2) Set the timer 8 looping value to the TM8CA register (the valid range: 1 to x'FFFF'). The TM8BC register counts from 0 to x'1FFF' when writing x'1FFF' to the TM8CA register.

TM8CA: x'00FE84'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 CA15	TM8 CA14	TM8 CA13	TM8 CA12	TM8 CA11	TM8 CA10	TM8 CA9	TM8 CA8	TM8 CA7	TM8 CA6	TM8 CA5	TM8 CA4	TM8 CA3	TM8 CA2	TM8 CA1	TM8 CA0
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1

(3) Set the timer 8 interrupt value to the TM8CB register (the valid range: 0 to TM8CA). Whenever the up or down counter reaches this value, a capture B interrupt occurs at the beginning of the next cycle.

TM8CB: x'00FE88'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 CB15	TM8 CB14	TM8 CB13	TM8 CB12	TM8 CB11	TM8 CB10	TM8 CB9	TM8 CB8	TM8 CB7	TM8 CB6	TM8 CB5	TM8 CB4	TM8 CB3	TM8 CB2	TM8 CB1	TM8 CB0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

If this setting is omitted, the binary counter may not count the first cycle. Do not change to any other operating modes.

(4) Set TM8NLD and TM8EN of the TM8MD register to 1 and 0 respectively. This enables TM8BC and RS.F.F.

(5) Set TM8NLD and TM8EN of the TM8MD register to 1. This starts the timer. Counting begins at the start of the next cycle.

■ Interrupt Enable Setting

(6) Enable interrupts after clearing all prior interrupt requests. To do this, set IQ0LV[2:0] of the external interrupt 0 control register (IQ0ICH) to the interrupt level 0 to 6), TM8BIR of the timer 8 capture B interrupt control register (TM8BICL) to 0, and TM8BIE of the timer 8 capture B interrupt control register (TM8BICH) to 1. Thereafter, a timer 8 capture B interrupt occurs when the TM8BC counter matches the TM8CB register.

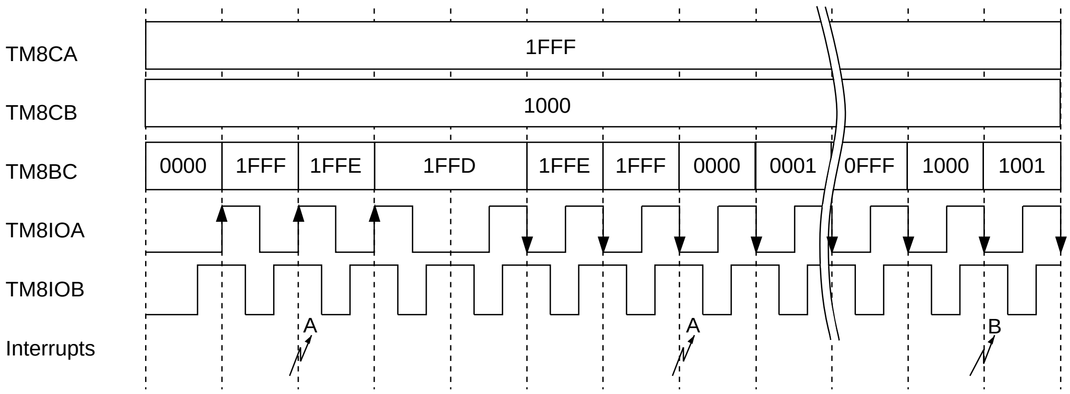
■ Interrupt Processing

(7) First, determine the interrupt group and factor, and clear TM8BIR flag during the interrupt service routine.

(8) Execute the interrupt service routine.

Timer 8 functions as a two-phase encoder input. Timer 8 does not operate when BOSC stops (in STOP mode). All external inputs are sampled on BOSC (synchronized with BOSC) when the external clock operates.

Figure 4-4-20 shows the counting direction. When counting down, the next value after 0 becomes the TM8CA value. When the TM8BC value matches the TM8CB value, a capture B interrupt occurs.



	Up Counting	Down Counting
TM8IOA	↓	↑
TM8IOB	1	1

Figure 4-4-20 Two-phase Encoder Input Timing (1x) (16-bit Timer)

4-4-8 One-shot Pulse Using 16-bit Timer

Timer 8 is used to generate a one-shot pulse. The pulse width is 2 cycles of BOSC/2.

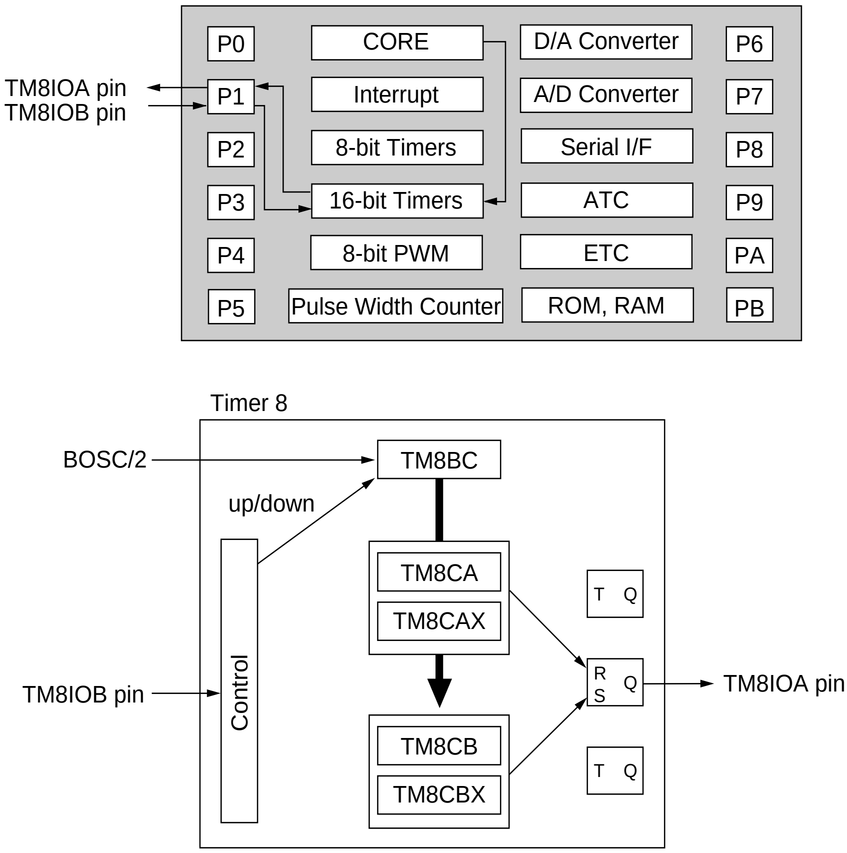


Figure 4-4-21 One-shot Pulse Output Block Diagram (16-bit Timer)

■ Timer 8 Setting

(1) Set the operating mode in the timer 8 mode register (TM8MD). Set counting stop. Select up counting. Select BOSC/2 as the clock source. Set the TM8BC count range to 0 to TM8CA. Select one-shot operation as the counter operating mode. Set the count start external trigger to start counting on the falling edge of TM8IOB pin.

Use the MOV instruction to set the data and only use 16-bit write operations.

The timer 8 binary counter (TM8BC) is stopped, and TM8BC register and RS.F.F. are initialized (cleared to 0).

TM8MD: x'00FE80'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 EN	TM8 NLD			TM8 UD1	TM8 UD0	TM8 TGE	TM8 ONE	TM8 MD1	TM8 MD0	TM8 ECLR	TM8 LP	TM8 ASEL	TM8 S2	TM8 S1	TM8 S0
0	0			0	0	1	1	0	0	0	1	1or0	0	1	1

(2) Set the pulse width. Since the width is 2 cycles of BOSC/2, set the timer 8 compare/capture register A (TM8CA) to 3 (the valid range is 1 to x'FFFE'). TM8BC counts from 0 to 3, and TM8IOA pin outputs 'H' while TM8BC counts from 2 to 3 by setting TM8CB in the next procedure (3). The operation is the same as that of the two-phase PWM output.

TM8CA: x'00FE84'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 CA15	TM8 CA14	TM8 CA13	TM8 CA12	TM8 CA11	TM8 CA10	TM8 CA9	TM8 CA8	TM8 CA7	TM8 CA6	TM8 CA5	TM8 CA4	TM8 CA3	TM8 CA2	TM8 CA1	TM8 CA0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1

(3) Write 1 to the timer 8 compare/capture register B (TM8CB). When TM8BC reaches TM8CB (TM8BC = 2), TM8IOA pin outputs 'H' at the start of the next cycle.

TM8CB: x'00FE88'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 CB15	TM8 CB14	TM8 CB13	TM8 CB12	TM8 CB11	TM8 CB10	TM8 CB9	TM8 CB8	TM8 CB7	TM8 CB6	TM8 CB5	TM8 CB4	TM8 CB3	TM8 CB2	TM8 CB1	TM8 CB0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

If this setting is omitted, the binary counter may not count the first cycle. Do not change to any other operating modes.

(4) Set TM8NLD and TM8EN of the TM8MD register to 1 and 0 respectively. This enables TM8BC and RS.F.F.

During the count operation, '1' is written automatically to TM8EN of the TM8MD register on the falling edge of TM8IOB pin. Therefore, counting starts at the beginning of the next cycle after TM8IOB pin falls. TM8EN of the TM8MD register can replace as the busy flag for one-shot operation.

Timer 8 generates a one-shot pulse. Timer 8 does not operate when BOSC stops (in STOP mode). All external inputs are sampled on BOSC (synchronized with BOSC) when the external clock operates.

Figure 4-4-22 shows the timing chart. Set TM8EN of the TM8MD register on the falling edge of TM8IOB pin, and counting starts at the beginning of the next cycle. Before counting starts, TM8BC is 0, the initial value of TM8IOA pin is 0, and R8 (reset) signal or S8 (set) signal cannot be output. (R8 and S8 are internal control signals.) When counting starts, the count changes from 0 to 1 and the S8 signal is output. This sets TM8IOA pin to 1 and outputs the one-shot pulse. When the count reaches 3, TM8BC resets from 3 to 0, and the R8 signal is output simultaneously. TM8IOA pin outputs 0. Since TM8ONE of the TM8MD register is set to 1, TM8EN of the TM8MD register is resets and then counting stops. The state is the same state before the falling edge of TM8IOB pin. When the falling edge of TM8IOB pin occurs again, set TM8EN of the TM8MD register, repeat the same operations and then results in the one-shot pulse output.

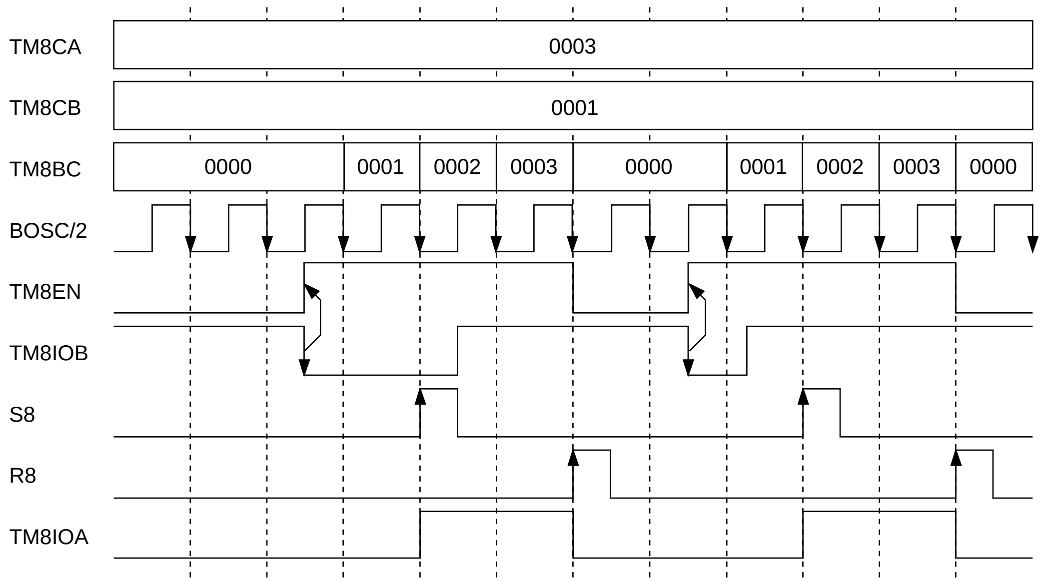
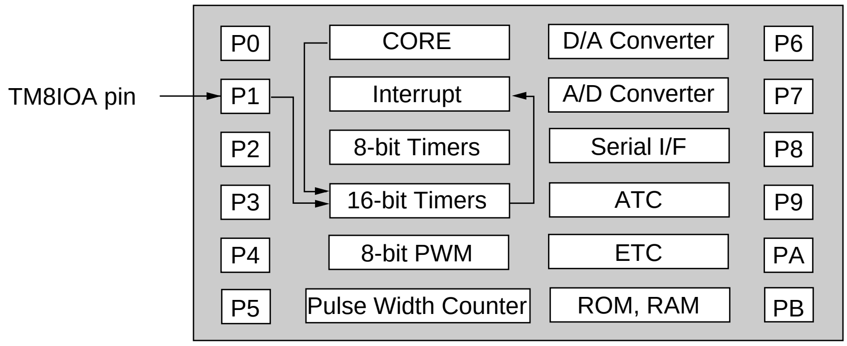


Figure 4-4-22 One-shot Pulse Output Timing (16-bit Timer)

4-4-9 External Count Direction Control Using 16-bit Timer

Timer 8 counts BOSC/2 and TM8IOA pin controls the count direction (up or down). An interrupt occurs when the counter reaches the value set in TM8CB register.



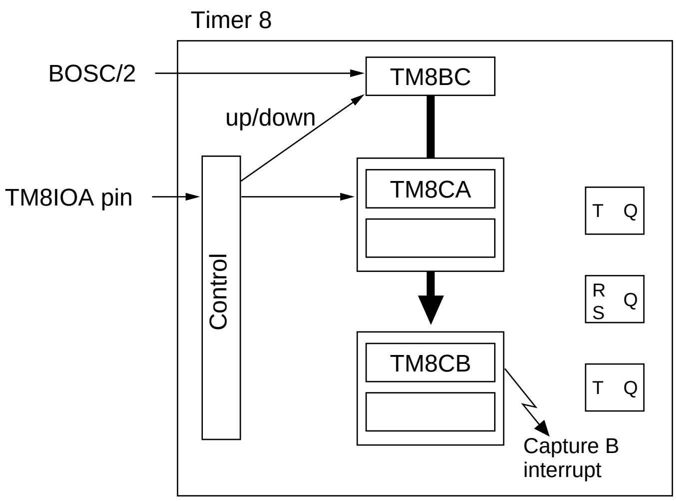


Figure 4-4-23 External Count Direction Control Block Diagram (16-bit Timer)

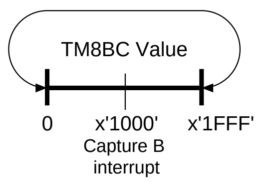


Figure 4-4-24 External Count Direction Control Configuration Example

Use the MOV instruction to set the data and only use 16-bit write operations.

The timer 8 binary counter (TM8BC) is stopped, and TM8BC register and RS.F.F. are initialized (cleared to 0).

■ Timer 8 Setting

(1) Set the operating mode in the timer 8 mode register (TM8MD). Set counting stop. The count direction is up when TM8IOA is 1, while the direction is down when TM8IOA is 0. Select BOSC/2 as the clock source. Set the TM8BC count range to 0 to TM8CA.

TM8MD: x'00FE80'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 EN	TM8 NLD			TM8 UD1	TM8 UD0	TM8 TGE	TM8 ONE	TM8 MD1	TM8 MD0	TM8 ECLR	TM8 LP	TM8 ASEL	TM8 S2	TM8 S1	TM8 S0
0	0			1	0	0	0	0	0	0	1	1or0	0	1	1

(2) Set the timer 8 looping value to the TM8CA register (the valid range: 1 to x'FFFE'). The TM8BC counter counts from 0 to x'1FFF' when writing x'1FFF' to the TM8CA register.

TM8CA: x'00FE84'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 CA15	TM8 CA14	TM8 CA13	TM8 CA12	TM8 CA11	TM8 CA10	TM8 CA9	TM8 CA8	TM8 CA7	TM8 CA6	TM8 CA5	TM8 CA4	TM8 CA3	TM8 CA2	TM8 CA1	TM8 CA0
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1

(3) Set the timer 8 interrupt value to the TM8CB register (the valid range: 0 to TM8CA). In this example, write x'1000'. Whenever the up or down counter reaches this value, a capture B interrupt occurs at the beginning of the next cycle.

TM8CB: x'00FE88'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 CB15	TM8 CB14	TM8 CB13	TM8 CB12	TM8 CB11	TM8 CB10	TM8 CB9	TM8 CB8	TM8 CB7	TM8 CB6	TM8 CB5	TM8 CB4	TM8 CB3	TM8 CB2	TM8 CB1	TM8 CB0
0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0

If this setting is omitted, the binary counter may not count the first cycle. Do not change to any other operating modes.

(4) Set TM8NLD and TM8EN of the TM8MD register to 1 and 0 respectively. This enables TM8BC and RS.F.F.

(5) Set TM8NLD and TM8EN of the TM8MD register to 1. This starts the timer. Counting begins at the start of the next cycle.

■ Interrupt Enable Setting

(6) Enable interrupts after clearing all prior interrupt requests. To do this, set IQ0LV[2:0] of the external interrupt 0 control register (IQ0ICH) to the interrupt level 0 to 6), TM8BIR of the timer 8 capture B interrupt control register (TM8BICL) to 0, and TM8BIE of the timer 8 capture B interrupt control register (TM8BICH) to 1. Thereafter, a timer 8 capture B interrupt occurs when the TM8BC counter matches the TM8CB register.

■ Interrupt Processing

(7) First, determine the interrupt group and factor, and clear TM8BIR flag during the interrupt service routine.

(8) Execute the interrupt service routine.

TM8IOA pin can control the timer 8 count direction. The count direction is controlled on the rising edge of the clock source (BOSC/2).

Timer 8 does not operate when BOSC stops (in STOP mode). All external inputs are sampled on BOSC (synchronized with BOSC) when the external clock operates.

Figure 4-4-25 shows the timing chart. In the example, an interrupt occurs when timer 8 changes from down counting to up counting.

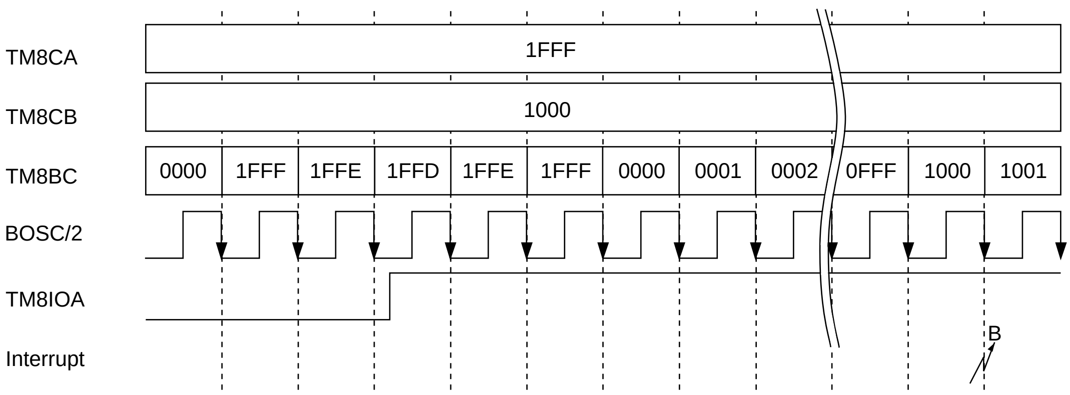


Figure 4-4-25 External Count Direction Control Timing (16-bit Timer)

4-4-10 External Reset Control Using 16-bit Timer

Timer 8 is reset by an external signal while counting up.

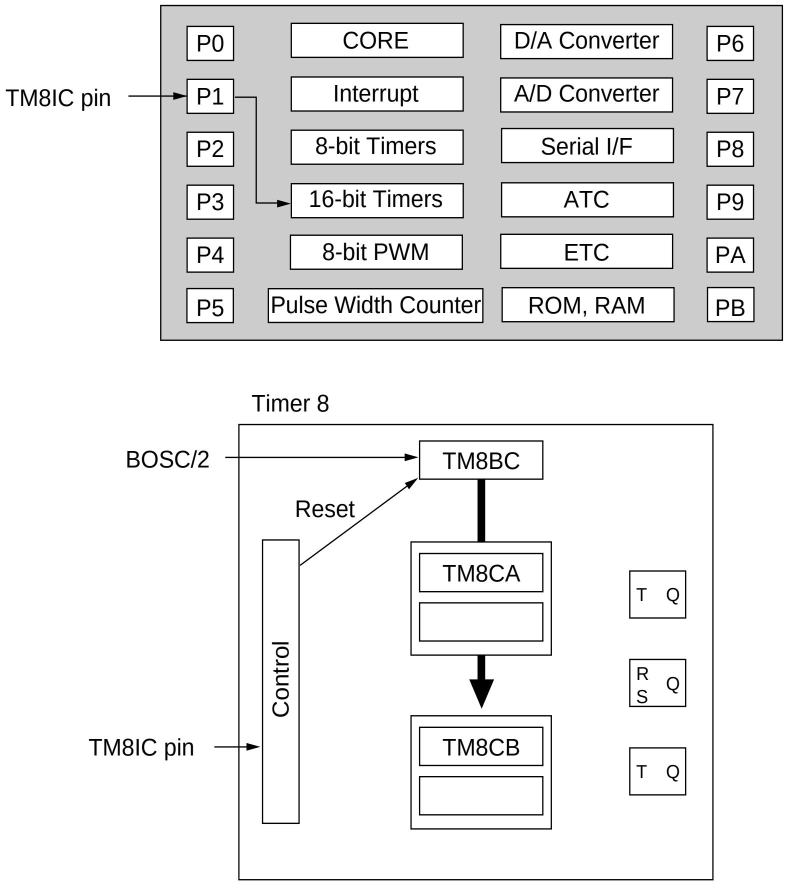


Figure 4-4-26 External Reset Control Block Diagram (16-bit Timer)

■ Timer 8 Setting

(1) Set the operating mode in the timer 8 mode register (TM8MD). Set counting stop. Select up counting. Select BOSC/2 as the clock source. Set the TM8BC clear condition to clear when TM8IC pin is high. Set the TM8BC count range to 0 to TM8CA.

Use the MOV instruction to set the data and only use 16-bit write operations.

TM8MD: x'00FE80'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 EN	TM8 NLD			TM8 UD1	TM8 UD0	TM8 TGE	TM8 ONE	TM8 MD1	TM8 MD0	TM8 ECLR	TM8 LP	TM8 ASEL	TM8 S2	TM8 S1	TM8 S0
0	0			0	0	0	0	0	0	1	1	1or0	0	1	1

The timer 8 binary counter (TM8BC) is stopped, and TM8BC register and RS.F.F. are initialized (cleared to 0).

(2) Set the timer 8 looping value to the TM8CA register (the valid range: 1 to x'FFFE'). The TM8BC register counts from 0 to x'1FFF' when writing x'1FFF' to the TM8CA register.

TM8CA: x'00FE84'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 CA15	TM8 CA14	TM8 CA13	TM8 CA12	TM8 CA11	TM8 CA10	TM8 CA9	TM8 CA8	TM8 CA7	TM8 CA6	TM8 CA5	TM8 CA4	TM8 CA3	TM8 CA2	TM8 CA1	TM8 CA0
0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1

(3) Set TM8NLD and TM8EN of the TM8MD register to 1 and 0 respectively. This enables TM8BC and RS.F.F.

(4) Set TM8NLD and TM8EN of the TM8MD register to 1. This starts the timer. Counting starts at the beginning of the next cycle.

If this setting is omitted, the binary counter may not count the first cycle. Do not change to any other operating modes.

Timer 8 is reset asynchronously while high is input from TM8IC pin. This allows to synchronize externally. It can be used to adjust the motor speed or to initialize the timer by the hardware.

Timer 8 does not operate when BOSC stops (in STOP mode). All external inputs are sampled on BOSC (synchronized with BOSC) when the external clock operates.

Figure 4-4-27 shows the timing chart.

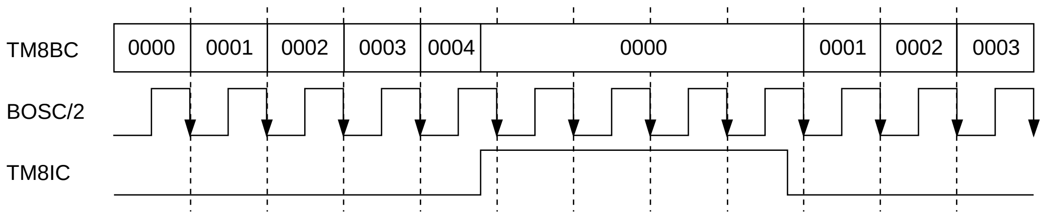


Figure 4-4-27 External Reset Control Timing (16-bit Timer)

4-5 Summary of 8-bit PWM Functions

4-5-1 Overview

The MN102H55D/55G/F55G has two 8-bit PWM waveform counters (timer 13 and timer 14). A timer can output two waveforms using two output pins.

BOSC/2 or timer 0 underflow is selected as the clock source. Each counter sets the PWM cycle. Each counter can output two PWM waveforms with the different duties since each counter has two output compare registers. The PWM counters can not generate interrupts.

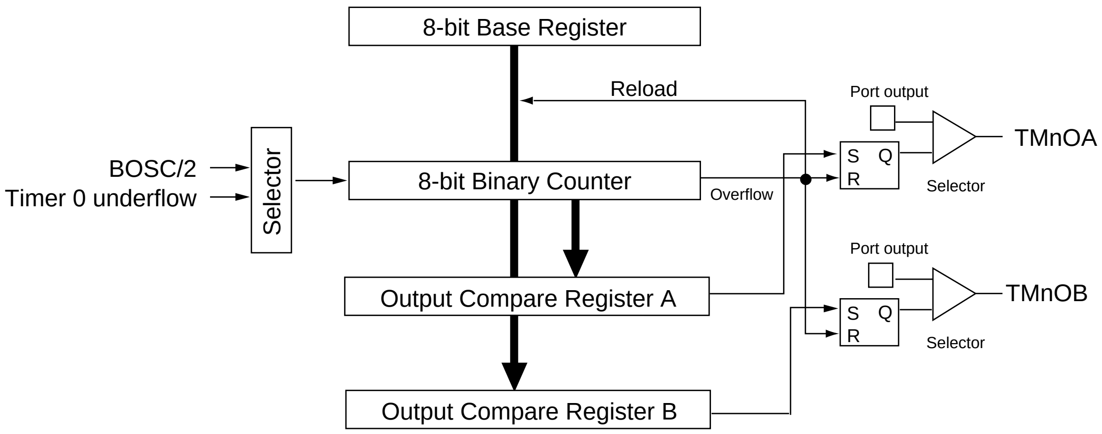


Figure 4-5-1 8-bit PWM Function

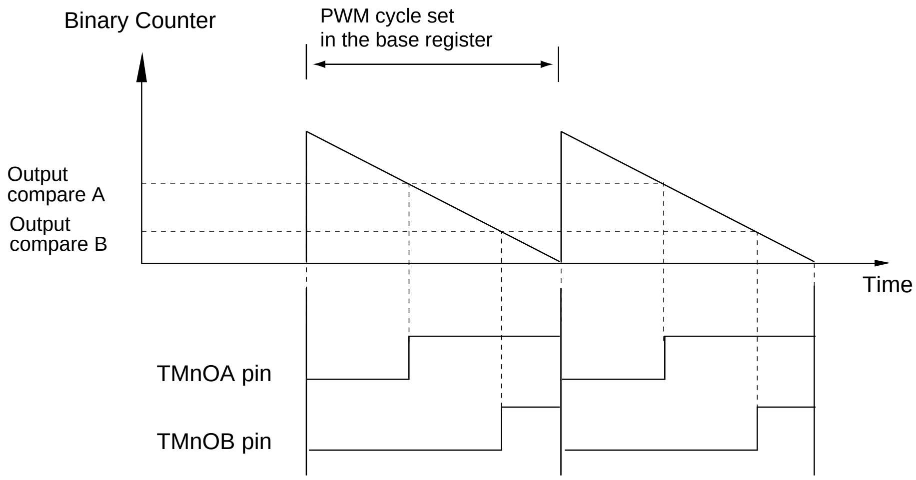


Figure 4-5-2 PWM Output Waveform

When the value in the output compare register matches the value in the binary counter, high level is output. When the binary counter underflows, low level is output. Counting starts when the value in the base register is read into the binary counter again. High and low signals are output by switching TMnOA pin or TMnOB pin to a general purpose port without using the PWM function. Figure 4-5-2 shows the waveforms which the PWM outputs. The PWM cycle is fixed and two waveforms with the different duties can be output. The binary counter is down counting.

Table 4-5-1 8-bit PWM Functions

	Timer 13	Timer 14
Clock Source	0: BOSC/2 1: Timer 0 underflow	0: BOSC/2 1: Timer 0 underflow
Output Compare Register	TM13CA TM13CB	TM14CA TM14CB
Output Pin	TM13OA TM13OB	TM14OA TM14OB

4-5-2 Control Registers

TMnMD

Mode Register

7	6	5	4	3	2	1	0
TMn EN	TMn LD	TMn CLR			TMn OB	TMn OA	TMn S

TMnS	Clock Source Selection
0	BOSC/2
1	Timer 0 underflow

TMnOA	TMnOA Output Edge Selection
0	Positive logic
1	Negative logic

TMnOB	TMnOB Output Edge Selection
0	Positive logic
1	Negative logic

TMnCLR	TMnBC and RS.F.F. Clear
0	No operation
1	Clear

TMnLD	Read TMnBR value to TMnBC
0	No operation
1	Read

TMnEN	TMnBC Counting Operation
0	Count stop
1	Count operation

TMnBC

7	6	5	4	3	2	1	0
TMn BC7	TMn BC6	TMn BC5	TMn BC4	TMn BC3	TMn BC2	TMn BC1	TMn BC0

Binary Counter

TMnBR

7	6	5	4	3	2	1	0
TMn BR7	TMn BR6	TMn BR5	TMn BR4	TMn BR3	TMn BR2	TMn BR1	TMn BR0

Base Register

TMnCA

7	6	5	4	3	2	1	0
TMn CA7	TMn CA6	TMn CA5	TMn CA4	TMn CA3	TMn CA2	TMn CA1	TMn CA0

Output Compare Register A

TMnCB

7	6	5	4	3	2	1	0
TMn CB7	TMn CB6	TMn CB5	TMn CB4	TMn CB3	TMn CB2	TMn CB1	TMn CB0

Output Compare Register B

Table 4-5-2 List of 8-bit PWM Registers

Register	Address	Function
TM13BC	x'00FE08'	Timer 13 Binary Counter
TM13CA	x'00FE0A'	Timer 13 Output Compare Register A
TM13BR	x'00FE18'	Timer 13 Base Register
TM13CB	x'00FE1A'	Timer 13 Output Compare Register B
TM13MD	x'00FE28'	Timer 13 Mode Register
TM14BC	x'00FE09'	Timer 14 Binary Counter
TM14CA	x'00FE0B'	Timer 14 Output Compare Register A
TM14BR	x'00FE19'	Timer 14 Base Register
TM14CB	x'00FE1B'	Timer 14 Output Compare Register B
TM14MD	x'00FE29'	Timer 14 Mode Register

4-6 8-bit PWM Setup Examples

4-6-1 8-bit PWM Output

Timer 13 is used to output PWM from TM13OA pin and TM13OB pin. The PWM cycle is timer 0 underflow/9. The TM13OA pin duty is 1:2, and the TM13OB pin duty is 2:1. The PWM output starts low. Set timer 0 to underflow at BOSC/2 divided by 2.

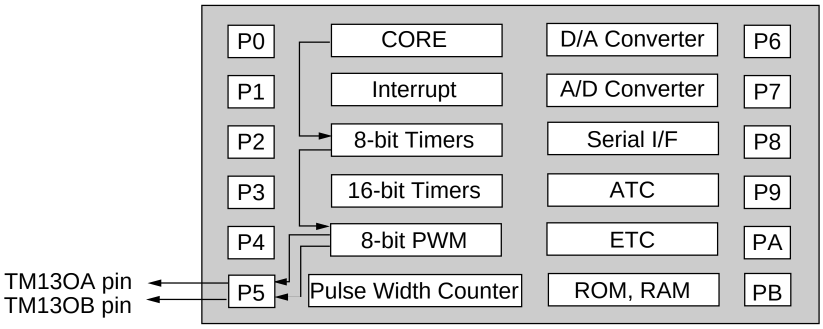


Figure 4-6-1 8-bit PWM Block Diagram

(1) Set the timer 0 divisor. Since timer 0 divides BOSC/2 by 2, set the timer 0 base register (TM0BR) to 1. (The valid range for TM0BR is 0 to 255, and the actual setting is the divisor to be set-1.)

TM0BR: x'00FE10'

7	6	5	4	3	2	1	0
TM0 BR7	TM0 BR6	TM0 BR5	TM0 BR4	TM0 BR3	TM0 BR2	TM0 BR1	TM0 BR0
0	0	0	0	0	0	0	1

(2) Load TM0BR value to the timer 0 binary counter (TM0BC). At the same time, select BOSC/2 as the clock source.

TM0MD: x'00FE20'

7	6	5	4	3	2	1	0
TM0 EN	TM0 LD					TM0 S1	TM0 S0
0	1					0	0

(3) Set TM0LD and TM0EN of the TM0MD register to 0 and 1 respectively. This starts the timer. Counting begins at the start of the next cycle. When the timer 0 binary counter reaches 0, the value 1 of the timer 0 base register is loaded automatically to the TM8BC counter at the next count.

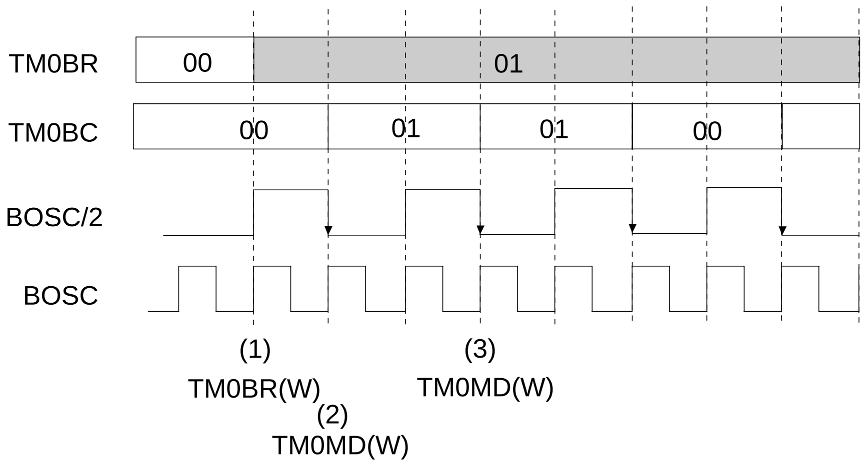


Figure 4-6-2 Timer 0 Timing

(4) Set the PWM cycle to the timer 13 base register. Since the PWM cycle is timer 0 underflow/9, set '8' to the TM13BR register.

TM13BR: x'00FE18'

7	6	5	4	3	2	1	0
TM13 BR7	TM13 BR6	TM13 BR5	TM13 BR4	TM13 BR3	TM13 BR2	TM13 BR1	TM13 BR0
0	0	0	0	1	0	0	0

(5) Set the duty to the timer 13 output compare register. When the TM13BC counter matches the TM13CA register, the PWM output of the TM13OA pin changes to low. When the TM13BC counter underflows, the PWM output of the TM13OA pin changes to high. When the TM13BC counter matches the TM13CB register, the PWM output of the TM13OB pin changes to low. When the TM13BC counter underflows, the PWM output of the TM13OB pin changes to high. The PWM output starts low at first. The TM13OA pin duty is 1:2, while the TM13OB pin duty is 2:1. Since the TM13BC counter is down counting, set the TM13CA register and the TM13CB register to 5 and 2 respectively.

TM13CA: x'00FE0A'

7	6	5	4	3	2	1	0
TM13 CA7	TM13 CA6	TM13 CA5	TM13 CA4	TM13 CA3	TM13 CA2	TM13 CA1	TM13 CA0
0	0	0	0	0	1	0	1

TM13CB: x'00FE1A'

7	6	5	4	3	2	1	0
TM13 CB7	TM13 CB6	TM13 CB5	TM13 CB4	TM13 CB3	TM13 CB2	TM13 CB1	TM13 CB0
0	0	0	0	0	0	1	0

(6) Load TM0BR value to the timer 13 binary counter (TM13BC). At the same time, select timer 0 underflow as the clock source. Set the PWM waveform polarity, which is output from TM13OA pin and TM13OB pin, to the positive logic. (When setting the polarity to the negative logic, an error of inverting high and low occurs.) To clear TM13BC counter or RS.F.F. for TM13OA pin output and TM13OB pin output, set the TM13CLR flag to 1.

TM13MD: x'00FE28'

7	6	5	4	3	2	1	0
TM13 EN	TM13 LD	TM13 CLR			TM13 OB	TM13 OA	TM13 S
0	1	1			0	0	1

(7) Set TM13OA pin and TM13OB pin. Since TM13OA pin and TM13OB pin function as P50 and P51 respectively, set the port 5 mode register L (P5LMD) to PWM output. Setting the port 5 direction control register (P5DIR) is not required. The PWM is output regardless of the P5DIR register value.

P5LMD: x'00FFF8'

7	6	5	4	3	2	1	0
P5 LMD7	P5 LMD6	P5 LMD5	P5 LMD4	P5 LMD3	P5 LMD2	P5 LMD1	P5 LMD0
0	0	0	0	1	0	1	0

(8) Set TM13LD, TM13EN and TM13CLR of the TM13MD register to 0, 1 and 0 respectively. This starts the timer. When the timer 0 binary counter reaches 0, the value 1 of the timer 0 base register is loaded automatically to the TM8BC counter at the next count.

TM13MD: x'00FE28'

7	6	5	4	3	2	1	0
TM13 EN	TM13 LD	TM13 CLR			TM13 OB	TM13 OA	TM13 S
1	0	0			0	0	1

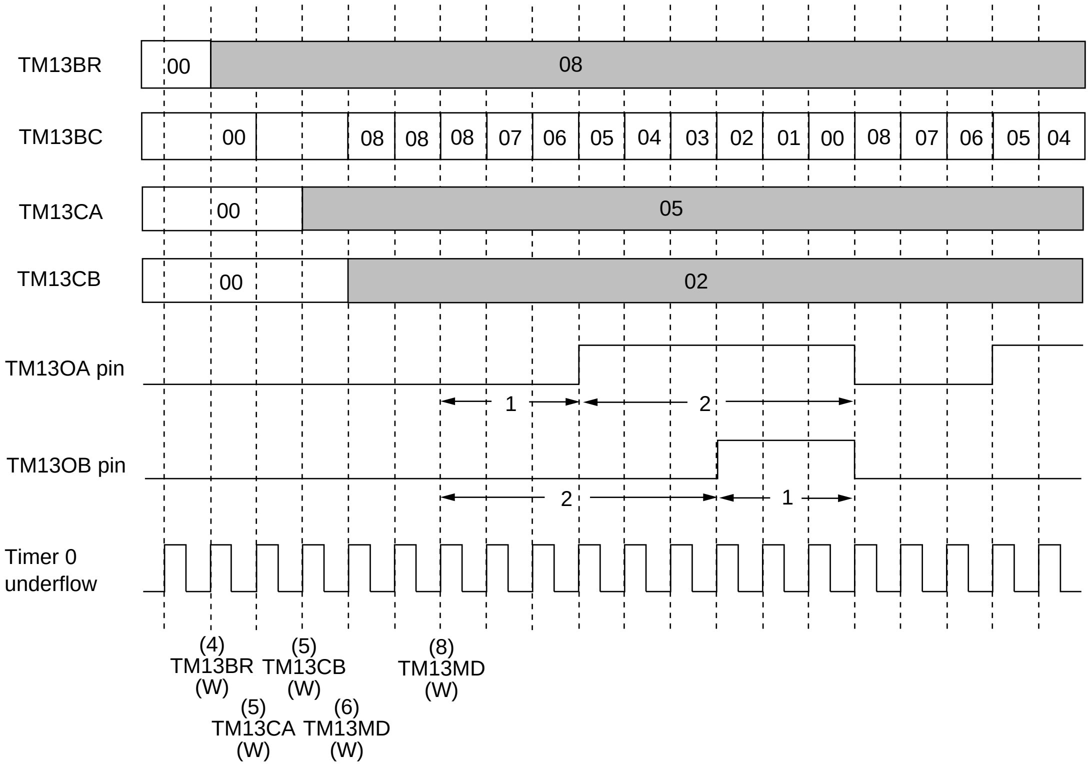


Figure 4-6-3 8-bit PWM Timing

4-7 16-bit Pulse Width Measure Functions

4-7-1 Overview

The MN102H55D/55G/F55G has one 16-bit pulse width measure counter.

The 16-bit binary counter value is read into the 16-bit capture register on the rising edge of the pulse waveform which inputs to TM15IA pin. Timer 0 underflow, TM15IB pin, BOSC/2 or BOSC is selected as the clock source.

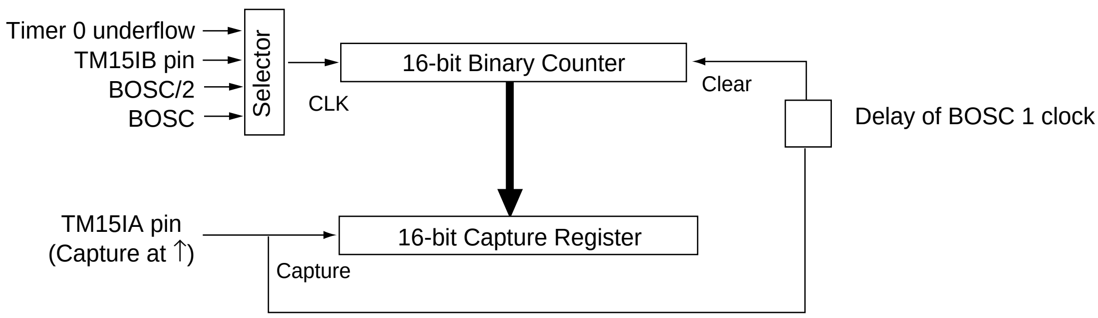


Figure 4-7-1 16-bit Pulse Width Measure Counter

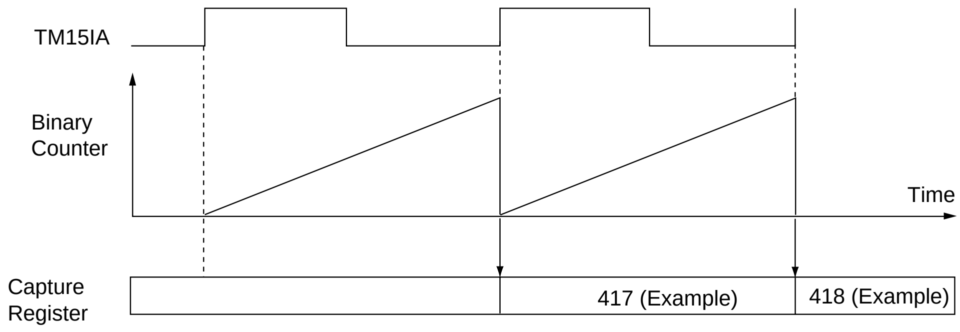


Figure 4-7-2 16-bit Pulse Width Measure Counter Operation Example

The binary counter is up counting. The contents of the binary counter are loaded into the capture register on the rising edge of TM15IA pin. The binary counter is cleared after BOSC 1 clock. The pulse width is always stored in the capture register.

4-7-2 Control Registers

TM15MD: x'00FED0'

TM15 EN														TM15 CLK1	TM15 CLK0	**Mode Register**

Timer 15 Count Control
0: Count stop
1: Counting

Clock Source Selection
00: Timer 0 underflow
01: TM15IB
10: BOSC/2
11: BOSC

TM15BC: x'00FED2'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM15 BC15	TM15 BC14	TM15 BC13	TM15 BC12	TM15 BC11	TM15 BC10	TM15 BC9	TM15 BC8	TM15 BC7	TM15 BC6	TM15 BC5	TM15 BC4	TM15 BC3	TM15 BC2	TM15 BC1	TM15 BC0

Binary Counter

TM15CA: x'00FED4'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM15 CA15	TM15 CA14	TM15 CA13	TM15 CA12	TM15 CA11	TM15 CA10	TM15 CA9	TM15 CA8	TM15 CA7	TM15 CA6	TM15 CA5	TM15 CA4	TM15 CA3	TM15 CA2	TM15 CA1	TM15 CA0

Capture Register

Table 4-7-1 List of 16-bit Pulse Width Measure Registers

Register	Address	Function
TM15MD	x'00FED0'	Timer 15 Mode Register
TM15BC	x'00FED2'	Timer 15 Binary Counter
TM15CA	x'00FED4'	Timer 15 Capture Register A

4-8 16-bit Pulse Width Counter Setup

4-8-1 16-bit Pulse Width Measure Counter

Timer 15 is used to measure the pulse width which is input from TM15IA pin. The pulse width is stored in the TM15CA register. Select TM15IB input as the clock source. Set the pulse width input from TM15IA pin to more than the width of the selected clock source. Set the pulse width input from TM15IB pin to more than BOSC/2.

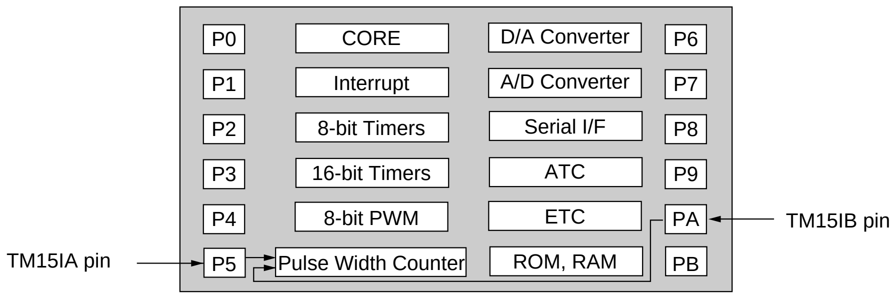


Figure 4-8-1 16-bit Pulse Width Measure Counter Block Diagram

(1) Set the timer 15 mode register (TM15MD). Select TM15IB pin input as the clock source. Set counting stop.

TM15MD: x'00FED0'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM15 EN														TM15 CLK1	TM15 CLK0
0														0	1

(2) Set TM15IA pin and TM15IB pin. By writing x'100' to P5HMD[4:2] flags of the port 5 mode register H (P5HMD), set P56 pin to TM15IA pin input. The P56 direction control becomes input automatically. Setting the port 5 direction control register (P5DIR) is invalid.

P5HMD: x'00FFF9'

7	6	5	4	3	2	1	0
			P5 HMD4	P5 HMD3	P5 HMD2	P5 HMD1	P5 HMD0
			1	0	0		

Set PA4 pin to TM15IB pin input by writing '1' to bit 4 of the port A mode register (PAMD). The PA4 pin direction control becomes input automatically. Setting the port A direction control register (PADIR) is invalid.

PAMD: x'00FFDC'

7	6	5	4	3	2	1	0
			PA MD4	PA MD3	PA MD2	PA MD1	PA MD0
			1				

(3) Set TM15EN flag of the TM15MD register to 1. This starts the timer.

TM15MD: x'00FED0'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM15 EN														TM15 CLK1	TM15 CLK0
1														0	1

Thereafter, the TM15BC counter starts counting up on the rising edge of TM15IB pin input. The TM15BC counter value is loaded to the timer 15 capture register (TM15CA) on the rising edge of TM15IA pin. When TM15IA pin rises, clear TM15BC counter after BOSC 1 clock. This operation allows to store the pulse width, which is input from TM15IA pin, to the TM15CA register.

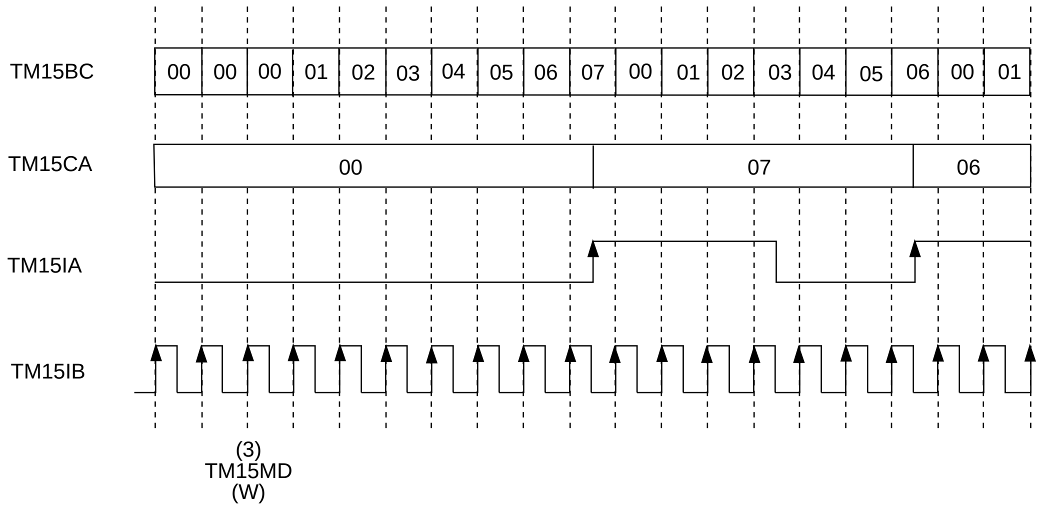


Figure 4-8-2 16-bit Pulse Width Measure Counter Timing

Chapter 5 Serial Interface

5

5-1 Serial Interface

5-1-1 Overview

The MN102H55D/55G/F55G contains two serial interfaces (serial 3 and serial 4) with asynchronous mode, clock synchronous mode and I^{2}C mode. It also has three serial interfaces (serial 0, serial 1 and serial 2) reserved for clock synchronous mode. The maximum baud rate in clock synchronous mode is 8.5 Mbps. The maximum baud rate in asynchronous mode is 28800 bps with a 34-MHz oscillator. (1228800 bps is possible by setting the oscillation frequency to 19.6608 MHz.)

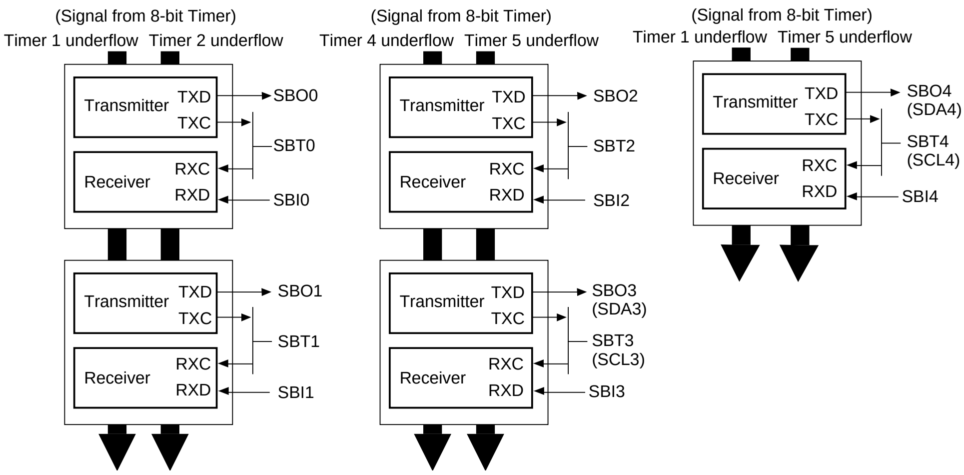


Figure 5-1-1 Serial Interface Configuration

Table 5-1-1 Serial Interface Functions

	Clock Synchronous Mode	Asynchronous Mode	I^2C Mode
Parity	None, 0, 1, Even, Odd		Master transmission and reception are possible. (No start sequence detection function)
Character Length	7-bit, 8-bit		
Bit Order	LSB first or MSB first(8-bit only)		
Clock Source	1/2, 1/8 of timers 1, 2, 4, 5 underflow External clock	1/8 of timers 1, 2, 4, 5 underflow	
Maximum Baud Rate	8.5 Mbps (with a 34-MHz oscillator)	28800 bps (1228800 bps) *	
Error Detection	Parity error Overrun error	Parity error Overrun error Framing error	
Buffers	Independent transmit/receive buffers (single tranmit buffer, double receive buffer)		
Interrupts	Transmission or reception end interrupt		

* When the oscillation frequency of 19.6608 MHz is selected.

5-1-2 Control Registers

Three registers control the serial interface: the serial transmit/receive buffers (SCnTRB), the serial port status registers (SCnSTR) and the serial control registers (SCnCTR).

SCnTRB

	7	6	5	4	3	2	1	0
	SCn TRB7	SCn TRB6	SCn TRB5	SCn TRB4	SCn TRB3	SCn TRB2	SCn TRB1	SCn TRB0
Reset	0	0	0	0	0	0	0	0

Transmission starts when the data is written to the SCnTRB register. The CPU reads the received data by reading the SCnTRB register. During 7-bit data reception, the MSB (bit 7) is set to 0. When an serial 0 reception end interrupt occurs, or the SCnRXA flag of the SCnSTR register is 1, the CPU can read the SCnTRB register.

SCnSTR

	7	6	5	4	3	2	1	0
	SCn TBY	SCn RBY	SCn ISP	SCn RXA	SCn IST	SCn FE	SCn PE	SCn OE
Reset	0	0	0	0	0	0	0	0

SCnOE	Overrun Error *1
0	No error
1	Error

SCnPE	Parity Error *2
0	No error
1	Error

SCnFE	Framing Error *3
0	No error
1	Error

SCnIST	I^2C Start Sequence Detect (cleared by read or write operation of SCnTRB)	
	For Serial 0, 1, 2	For Serial 3, 4
0	Reserved (set to 0)	No start sequence detected
1		Start sequence detected

SCnRXA	Received Data
0	No received data
1	Received data

SCnISP	I^2C Stop Sequence Detect (cleared by read or write operation of SCnTRB)	
	For Serial 0, 1, 2	For Serial 3, 4
0	Reserved (set to 0)	No stop sequence detected
1		Stop sequence detected

SCnRBY	Reception Busy Flag *4
0	Ready to receive
1	Reception in progress

SCnTBY	Transmission Busy Flag
0	Ready to transmit
1	Transmission in progress

*1 An overrun error occurs when the next data is received completely before the CPU reads the received data (SCnTRB). Overrun error data is updated whenever the last data bit (seventh or eighth bit) is received.

*2 A parity error occurs when the parity bit is 1 although it is set to 0, when the parity bit is 0 although it is set to 1, when the parity bit is odd although it is set to even, and when the parity bit is even although it is set to odd. Parity error data is updated whenever the parity bit is received.

*3 A framing error occurs when the stop bit is 0. Framing error data is updated whenever the stop bit is received.

*4 Do not use the SCnRBY flag to set polling for the received data wait in clock synchronous mode. Use the interrupt service routine, the serial interrupt flag or the SCnRXA flag.

SCnCTR

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SCnTEN	SCnREN	SCnBRE	SCnIIC	SCnPTL	—	SCnOD	SCnICM	SCnLN	SCnPTY2	SCnPTY1	SCnPTY0	SCnSB	—	SCnS1	SCnS0
Reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SCnS[1:0]	Clock Source Selection *		
	For Serial 0, 1	For Serial 2, 3	For Serial 4
00	SBTn pin	SBTn pin	SBTn pin
01	Timer 1 underflow (1/8)	Timer 4 underflow (1/8)	Timer 1 underflow (1/8)
10	Timer 2 underflow (1/2)	Timer 5 underflow (1/2)	Timer 5 underflow (1/2)
11	Timer 2 underflow (1/8)	Timer 5 underflow (1/8)	Timer 5 underflow (1/8)

* Some selections cannot be made in each mode (asynchronous/synchronous/I C).
Please refer to list of special registers in appendix for setting. Always select 1/8 of clock in asynchronous mode.

SCnSB	Stop Bit Selection (asynchronous mode)	SBO Output Hold Time (clock synchronous mode by SBTn pin)
0	1-bit	More than BOSC cycles
1	2-bit	More than Timer 1 or 4 underflow cycles

SCnPTY[2:0]	Parity Bit Selection
000	None
100	0 (output low)
101	1 (output high)
110	Even (1s are even)
111	Odd (1s are odd)
Others	Reserved

SCnLN	Character Length
0	7-bit
1	8-bit

SCnICM	I^2C Mode Selection	
	For Serial 0, 1, 2	For Serial 3, 4
0	Reserved (set to 0)	I^2C mode off
1		I^2C mode on

SCnOD	Bit Order Selection
0	LSB first
1	MSB first (select only when the character length is 8-bit.)

SCnPTL	Protocol Selection
0	Asynchronous mode
1	Clock synchronous mode, I^2C mode

SCnIIC	I^2C Start or Stop Sequence Output	
	For Serial 0, 1, 2	For Serial 3, 4
0	Reserved (set to 0)	Stop sequence output when changing from 1 to 0
1		Start sequence output when changing from 0 to 1

SCnBRE	Break Transmission
0	Don't break
1	Break

SCnREN	Receive Enable
0	Disable
1	Enable

SCnTEN	Transmit Enable
0	Disable
1	Enable

Table 5-1-2 List of Serial Interface Control Registers

Register		Address	R/W	Function
Serial 0	SC0CTR SC0TRB SC0STR	x'00FD80' x'00FD82' x'00FD83'	R/W R/W R	Serial 0 Control Register Serial 0 Transmit/Receive Buffer Serial 0 Status Register
Serial 1	SC1CTR SC1TRB SC1STR	x'00FD88 x'00FD8A' x'00FD8B'	R/W R/W R	Serial 1 Control Register Serial 1 Transmit/Receive Buffer Serial 1 Status Register
Serial 2	SC2CTR SC2TRB SC2STR	x'00FD90' x'00FD92' x'00FD93'	R/W R/W R	Serial 2 Control Register Serial 2 Transmit/Receive Buffer Serial 2 Status Register
Serial 3	SC3CTR SC3TRB SC3STR	x'00FD98 x'00FD9A' x'00FD9B'	R/W R/W R	Serial 3 Control Register Serial 3 Transmit/Receive Buffer Serial 3 Status Register
Serial 4	SC4CTR SC4TRB SC4STR	x'00FDA0' x'00FDA2' x'00FDA3'	R/W R/W R	Serial 4 Control Register Serial 4 Transmit/Receive Buffer Serial 4 Status Register

5-1-3 Serial Interface Connection

In half-duplex connection mode, the SBT pin requires a pullup resistor externally or an internal pullup resistor.

[Clock Synchronous Mode]

The serial interface can connect using either simplex or duplex synchronous transfer.

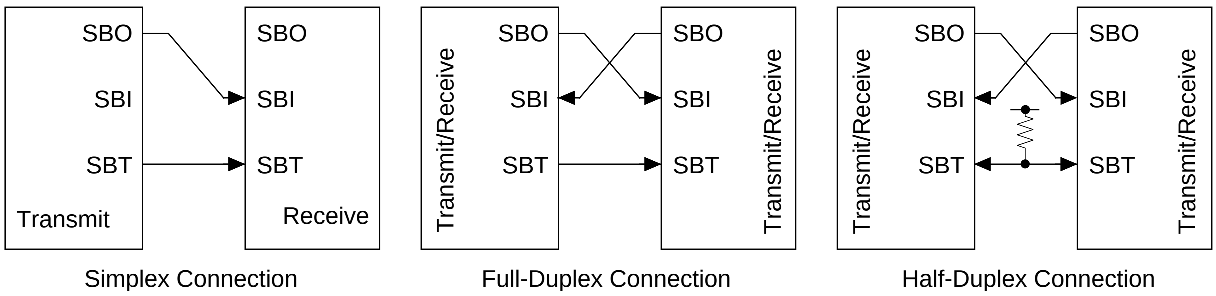


Figure 5-1-2 Synchronous Mode Connections

[Asynchronous Mode]

In the duplex (half-duplex) asynchronous mode, both SBT pins become input when they are not selected to transmit, so they required pullup resistors.

The serial interface can connect using either simplex or duplex asynchronous transfer.

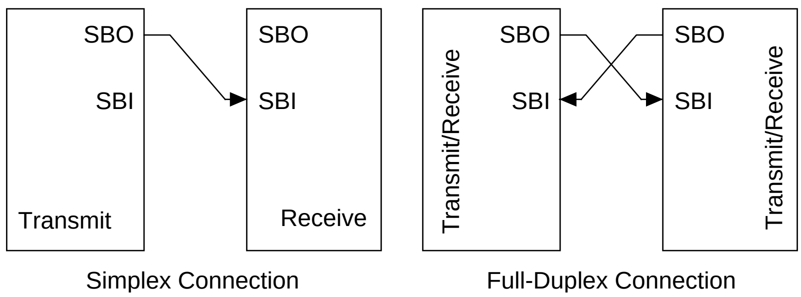


Figure 5-1-3 Asynchronous Mode Connections

[I²C Mode]

The SDA and SCL pins connect a pullup resistor externally or an internal pullup resistor by setting the register.

The serial interface can connect to slave transmitters or slave receivers.

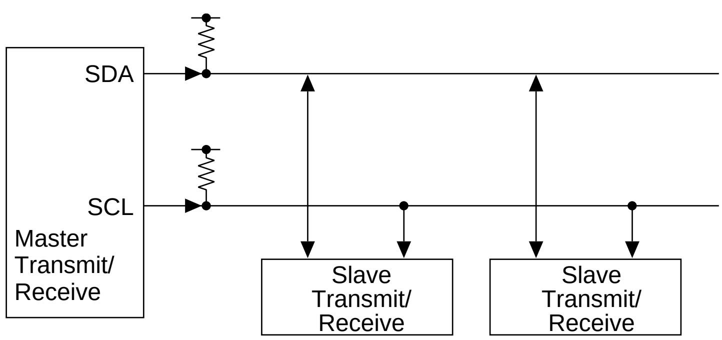


Figure 5-1-4 I²C Mode Connection

Table 5-1-3 Baud Rate Setting Example in Asynchronous Mode

When BOSC = 15 MHz

Baud Rate	Timer 5 Divisor	Timer 4 Divisor	Timer 1 Divisor
19200	49	Unused	Unused
9600	98	Unused	Unused
4800	98	Unused	2
2400	98	Unused	4
1200	781		Unused
600	1562		Unused
300	3125		Unused

$$\text{Baud Rate} = \frac{\text{BOSC (Hz)}}{16 \times (\text{Timer Divisor})}$$

Transmission/reception is possible within +/- 2 % of baud rate errors.

Asynchronous Serial Timing Charts

8-bit charater length, no parity, two stop bits

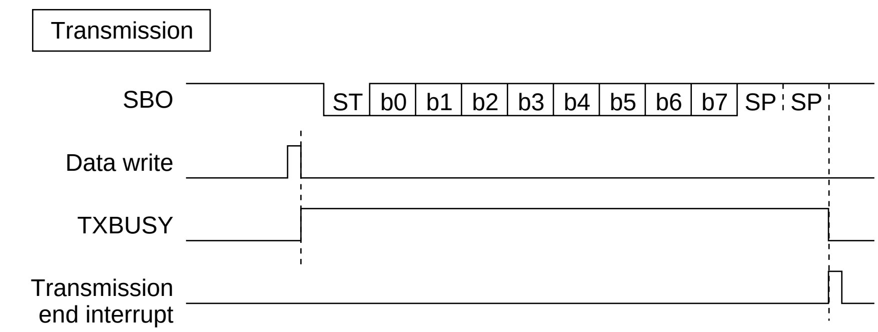


Figure 5-1-5 Asynchronous Serial Timing (Transmission)

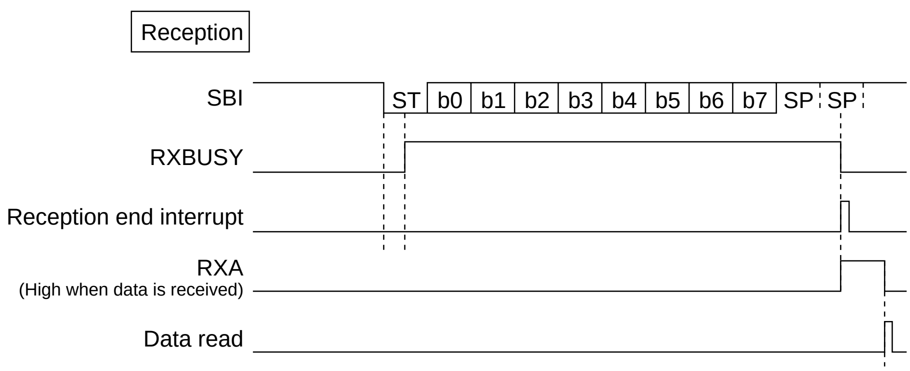


Figure 5-1-6 Asynchronous Serial Timing (Reception)

Synchronous Serial Timing Charts

8-bit charater length, parity

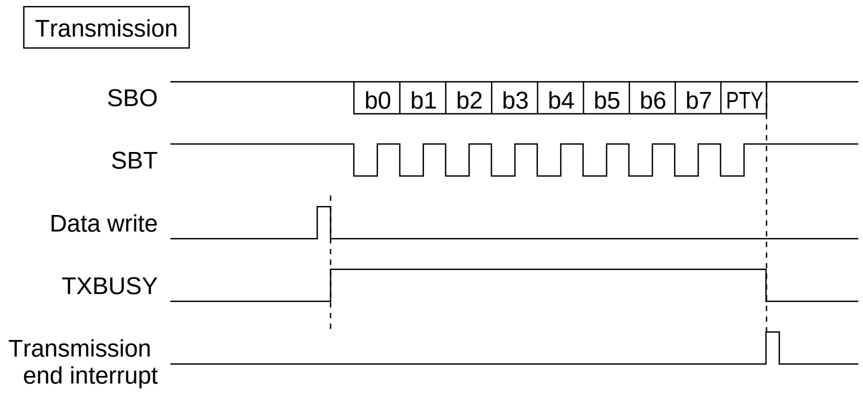


Figure 5-1-7 Synchronous Serial Timing (Transmission)

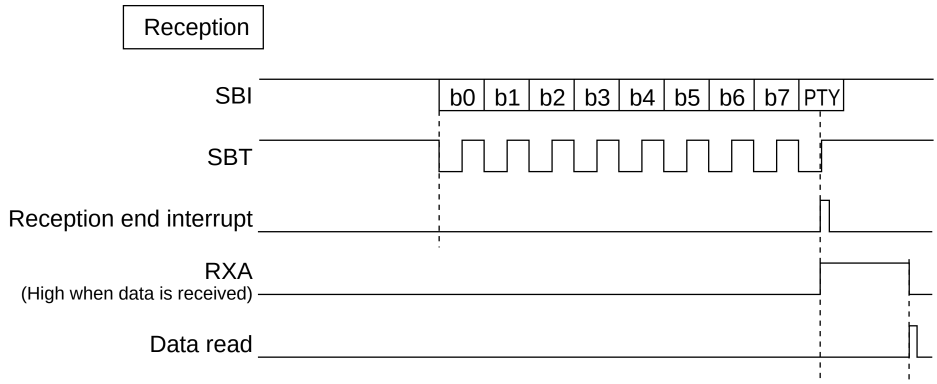


Figure 5-1-8 Synchronous Serial Timing (Reception)

5-2 Serial Interface Setup Examples

5-2-1 Serial Transmission in Asynchronous Mode

This section describes the example of serial interface 3 transmission in asynchronous mode with the following settings:

- Baud rate = 19200 bps (set transmit clock by timer 5)
- 8-bit data transmission
- two stop bits
- odd parity

Use a 8-bit timer to set the transmit clock.

See "5-2-3 Serial Clock Operation Example".

The next data is transmitted when a transmission end interrupt occurs.

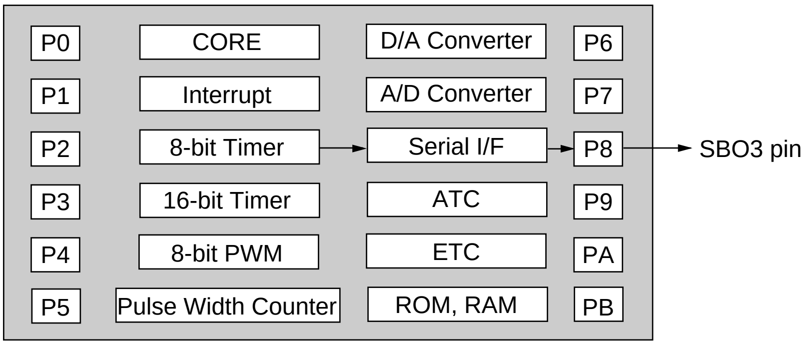


Figure 5-2-1 Asynchronous Transmission Block Diagram

Data transmission starts when the data is written to the serial 3 transmit/receive buffer (SC3TRB). The transmission starts synchronizing with timer 5 underflow. When an interrupt occurs, the SC3TRB register is cleared and then the next data is written to the SC3TRB register. If polling, the data must be written to the SC3TRB register after verifying that the SC3TBY flag of the serial 3 status register (SC3STR) is 0.

Setting the P8MMD to SBO3 output slects the P84 direction to output. Setting the port 8 I/O control register (P8DIR) is not required. P8DIR operates only when it is used as the port input or output.

■ Port Setting

(1) Set P8MMD[4:2] flags of the port 8 mode control register (P8MMD) to '011'. This setting allows to output SBO0 of serial interface.

P8MMD: x'00FFFD'

7	6	5	4	3	2	1	0
P8 MMD7	P8 MMD6	P8 MMD5	P8 MMD4	P8 MMD3	P8 MMD2	P8 MMD1	P8 MMD0
0	0	0	0	1	1	0	0

■ Serial Interface Setting

(1) Select timer 5 underflow (1/8) as the serial 3 clock source because the transfer base clock is 1/8 of timer 5. Select asynchronous mode, odd parity, two stop bits, 8-bit transmission and LSB first bit order. Set the SC3REN and SC3TEN flags of the serial 3 control register (SC3CTR) to disable and the reserved flags to 0.

SC3CTR: x'00FD98'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SC3 TEN	SC3 REN	SC3 BRE	SC3 IIC	SC3 PTL	-	SC3 OD	SC3 ICM	SC3 LN	SC3 PTY2	SC3 PTY1	SC3 PTY0	SC3 SB	-	SC3 S1	SC3 S0
0	0	0	0	0		0	0	1	1	1	1	1		1	1

(2) Enable serial transmission. To do this, set the SC3TEN flag of the serial 3 control register (SC3CTR) to 1.

SC3CTR: x'00FD98'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SC3 TEN	SC3 REN	SC3 BRE	SC3 IIC	SC3 PTL	-	SC3 OD	SC3 ICM	SC3 LN	SC3 PTY2	SC3 PTY1	SC3 PTY0	SC3 SB	-	SC3 S1	SC3 S0
1	0	0	0	0		0	0	1	1	1	1	1		1	1

(3) Enable interrupts after clearing all existing interrupt requests. At the same time, set the interrupt level. Thereafter, a serial transmission end interrupt occurs when the data transfer ends.

SC3TICL: x'00FC9C'

7	6	5	4	3	2	1	0
-	-	-	SC3T IR	-	-	-	SC3T ID
			0				0

SC2TICH: x'00FC98'

7	6	5	4	3	2	1	0
-	SC2T LV2	SC2T LV1	SC2T LV0	-	-	-	SC2T IE
	1	0	1				0

SC3TICH: x'00FC9D'

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	SC3T IE
0	0	0	0	0	0	0	1

The interrupt level is 5 in this example.

(4) Load the first transfer data to the serial 3 transmit/receive register (SC3TRB). Once the data is loaded to the SC3TRB register, transmission starts synchronizing with timer 5.

(5) Execute the interrupt service routine when a serial transmission end interrupt occurs. The interrupt service routine determines the interrupt group and vector and clears the SC3TIR flag.

(6) Write the next data. After the data is written, transfer starts in 1 - 2 cycles of the transfer base clock (timer 5 underflow).

Figure 5-2-2 illustrates the timing of asynchronous transmission.

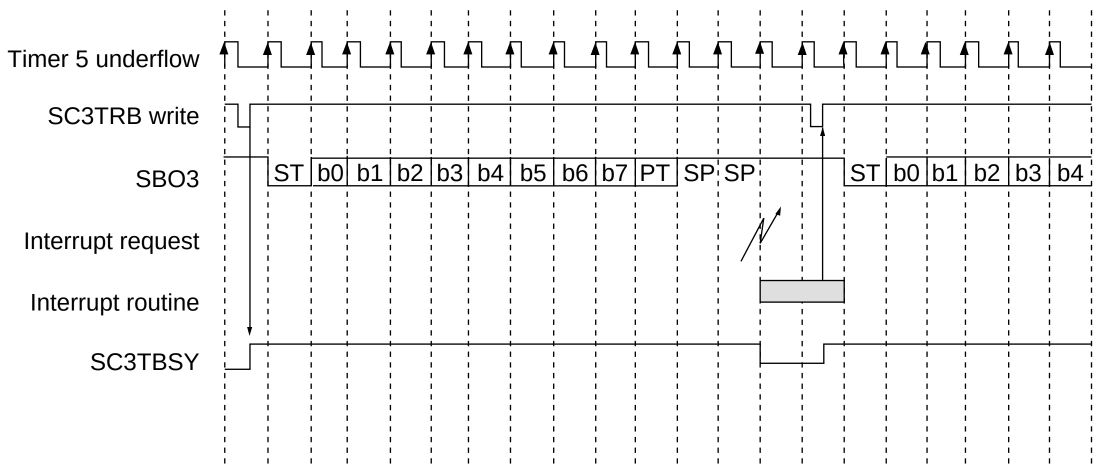


Figure 5-2-2 Serial 3 Asynchronous Transmission Timing

5-2-2 Serial Reception in Synchronous Mode

This section describes the example of serial interface 0 reception in synchronous mode with the following settings:

- LSB first bit order
- 8-bit data transfer
- odd parity

The data is received when a serial reception end interrupt occurs.

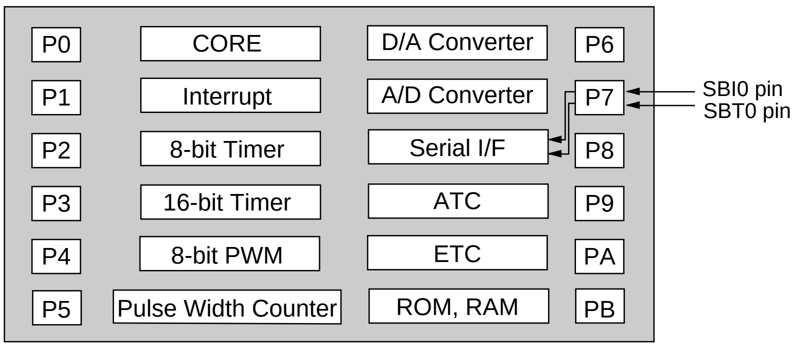


Figure 5-2-3 Synchronous Reception Block Diagram

In synchronous mode, the data input from the SBIn pin is received synchronizing with the SBTn pin and the received data is stored into the serial n transmit/receive buffer (SCnTRB). The SBTn clock is generated in transmitter or receiver. When the SBTn clock is generated in transmitter, the clock is transferred to the receiver through the SBTn pin as soon as the transmitted data is written to the SCnTRB register. On the other hand, when the SBTn clock is generated in receiver, the dummy data must be written to the SCnTRB register in the receiver after writing the transmitted data into the SCnTRB register in the transmitter. The reason for the dummy data requirement is because the clock is generated as soon as the data is written to the SCnTRB register.

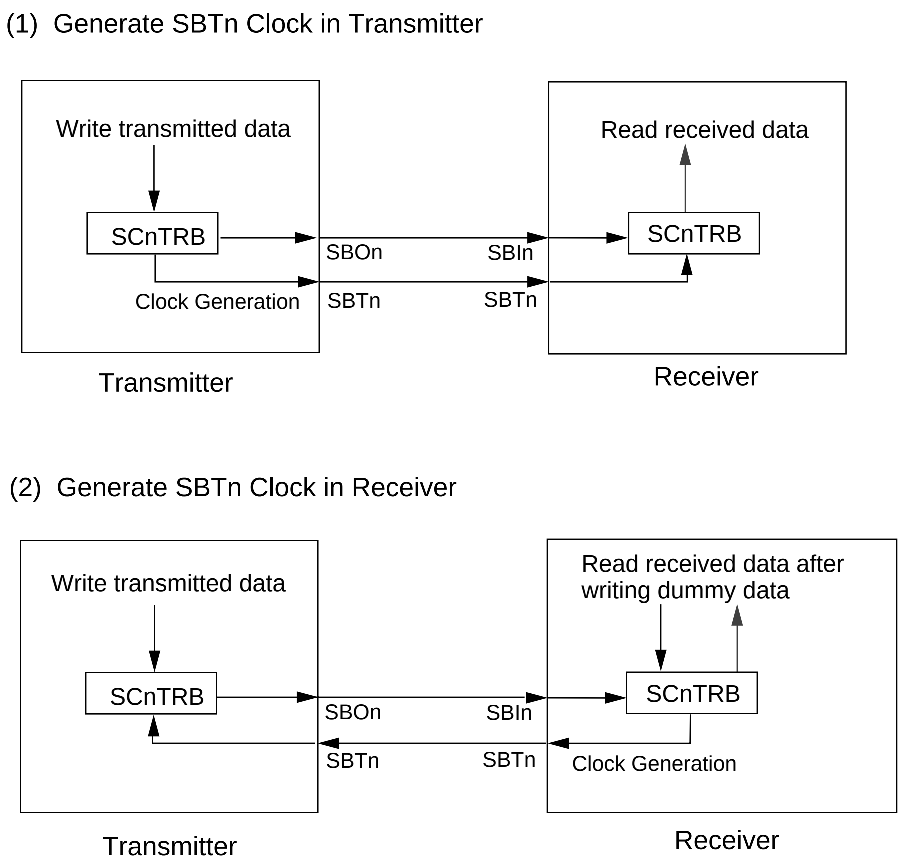


Figure 5-2-4 Clock Generation in Synchronous Reception

■ **Port 7 Setting**

(1) Set P7LMD[2:0] flags and P7LMD[4:3] flags of the port 7 mode register (P7LMD) to '001' and '01' respectively. This setting allows to input SBT0 and SBI0 of serial interface.

P7LMD: x'00FFFA'

7	6	5	4	3	2	1	0
P7 LMD7	P7 LMD6	P7 LMD5	P7 LMD4	P7 LMD3	P7 LMD2	P7 LMD1	P7 LMD0
0	0	0	0	1	0	0	1

■ **Serial Interface 0 Setting**

(1) Set the operating conditions in the serial 0 control register (SC0CTR). Select SBT0 pin as the clock source, 8-bit data transfer, odd parity, and reception enable. Set the reserved flags of the serial 0 control register (SC0CTR) to 0.

SC0CTR: x'00FD80'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SC0 TEN	SC0 REN	SC0 BRE	reser ved	reser ved	-	SC0 OD	reser ved	SC0 LN	SC0 PTY2	SC0 PTY1	SC0 PTY0	SC0 SB	-	SC0 S1	SC0 S0
0	1	0	0	0	-	0	0	1	1	1	1	0	-	0	0

(2) Enable interrupts after clearing all existing interrupt requests. At the same time, set the interrupt level. Thereafter, a serial reception end interrupt occurs when the data transfer ends.

SC0RICL: x'00FC92'

7	6	5	4	3	2	1	0
-	-	-	SC0R IR	-	-	-	SC0R ID
			0				0

SC0RICH: x'00FC93'

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	SC0R IE
							1

Thereafter, an interrupt occurs when the serial data is received.

5-2-3 Serial Clock Operation Example

This section describes how to set 19200 bps transfer clock for asynchronous serial interface by using timer 0 and timer 5 to divide BOSC/2 by 98. In this example, select 1/8 as the serial clock source and 8 times of baud rate as the transfer clock.

The serial Interface determines the baud rate with the 8-bit timer underflow. Select the transfer clock to make the timer 5 underflow twice or eight times of the baud rate. The serial interface divides the timer underflow by 2 or 8. (Always select 1/8 in asynchronous mode.) For a baud rate of 19200 bps, since BOSC/2 = 15 MHz with a 30-MHz oscillator,

15 MHz/98/8 = 19132.65 bps

This means the timer 5 underflow is divided by 98.
In this example, timer 0 is divided by 49 and timer 5 by 2.

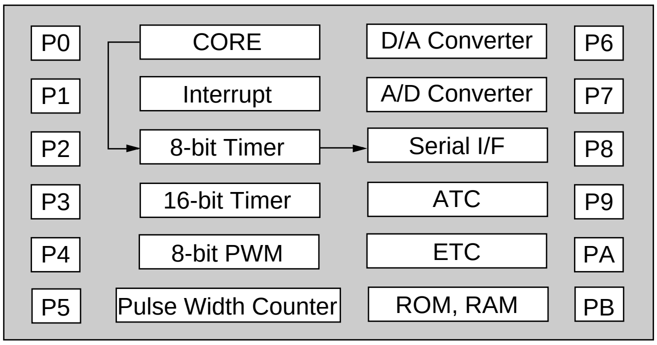


Figure 5-2-5 Serial Clock Block Diagram

Table 5-2-1 Transfer Clock Setup Example

Transfer Clock Setting Examples	Divisor at 30 MHz	Divisor Setting Method
38400 bps	49	Set divisor of 49 using timer 5.
19200 bps	98	Set divisor of 98 using timer 5. Set divisor of 49 using timer 0 and divisor of 2 using timer 5.
9600 bps	196	Set divisor of 196 using timers 4 and 5. Set divisor of 49 using timer 0 and divisor of 4 using timer 5.

■ Timer 0 Setting

This setting is unnecessary after a reset.

(1) Set timer 0 counting stop with the timer 0 mode register (TM0MD).

TM0MD: x'00FE20'

7	6	5	4	3	2	1	0
TM0 EN	TM0 LD					TM0 S1	TM0 S0
0							

(2) Set the timer 0 divisor. Since timer 0 divides BOSC/2 by 49, set the timer 0 base register (TM0BR) to 48 (x'30'). (The valid range for TM0BR is 0 to 255.)

TM0BR: x'00FE10'

7	6	5	4	3	2	1	0
TM0 BR7	TM0 BR6	TM0 BR5	TM0 BR4	TM0 BR3	TM0 BR2	TM0 BR1	TM0 BR0
0	0	1	1	0	0	0	0

Setting TM0EN and TM0LD to 0 is required between (3) and (4) in the bank address version and the linear address version, but this setting is not required in the linear address high-speed version.

(3) Load TM0BR value to the timer 0 binary counter (TM0BC). At the same time, select BOSC/2 as the clock source.

TM0MD: x'00FE20'

7	6	5	4	3	2	1	0
TM0 EN	TM0 LD					TM0 S1	TM0 S0
0	1					0	0

Do not change the clock source once you have selected it. Selecting the clock source while setting the count operation control will corrupt the value in the binary counter.

(4) Set TM0LD and TM0EN of the TM0MD register to 0 and 1 respectively. This starts the timer. Counting begins at the start of the next cycle.

When the timer 0 binary counter reaches 0 and loads the value 1 from the timer 0 base register at the next count, a timer 0 underflow interrupt request will be sent to the CPU.

■ Timer 5 Setting

(5) Set timer 5 counting stop with the timer 5 mode register (TM5MD).

This setting is unnecessary after a reset.

TM5MD: x'00FE25'

7	6	5	4	3	2	1	0
TM5 EN	TM5 LD					TM5 S1	TM5 S0
0							

(6) Set the timer 5 divisor. Since timer 5 divides BOSC/2 by 2, set the timer 5 base register (TM5BR) to 1. (The valid range for TM5BR is 0 to 255.)

TM5BR: x'00FE15'

7	6	5	4	3	2	1	0
TM5 BR7	TM5 BR6	TM5 BR5	TM5 BR4	TM5 BR3	TM5 BR2	TM5 BR1	TM5 BR0
0	0	0	0	0	0	0	1

(7) Load TM5BR value to the timer 5 binary counter (TM5BC). At the same time, select the timer 0 underflow as the clock source.

Setting TM5EN and TM5LD to 0 is required between (7) and (8) in the bank address version and the linear address version, but this setting is not required in the linear address high-speed version.

TM5MD: x'00FE25'

7	6	5	4	3	2	1	0
TM5 EN	TM5 LD					TM5 S1	TM5 S0
0	1					0	1

(8) Set TM5LD and TM5EN of the TM5MD register to 0 and 1 respectively. This starts the timer. Counting begins at the start of the next cycle.

Do not change the clock source once you have selected it. Selecting the clock source while setting the count operation control will corrupt the value in the binary counter.

When the timer 5 binary counter reaches 0 and loads the value 1 from the timer 5 base register at the next count, a timer 5 underflow interrupt request will be sent to the CPU.

The serial interface operates synchronizing with the timer 5 underflow output.

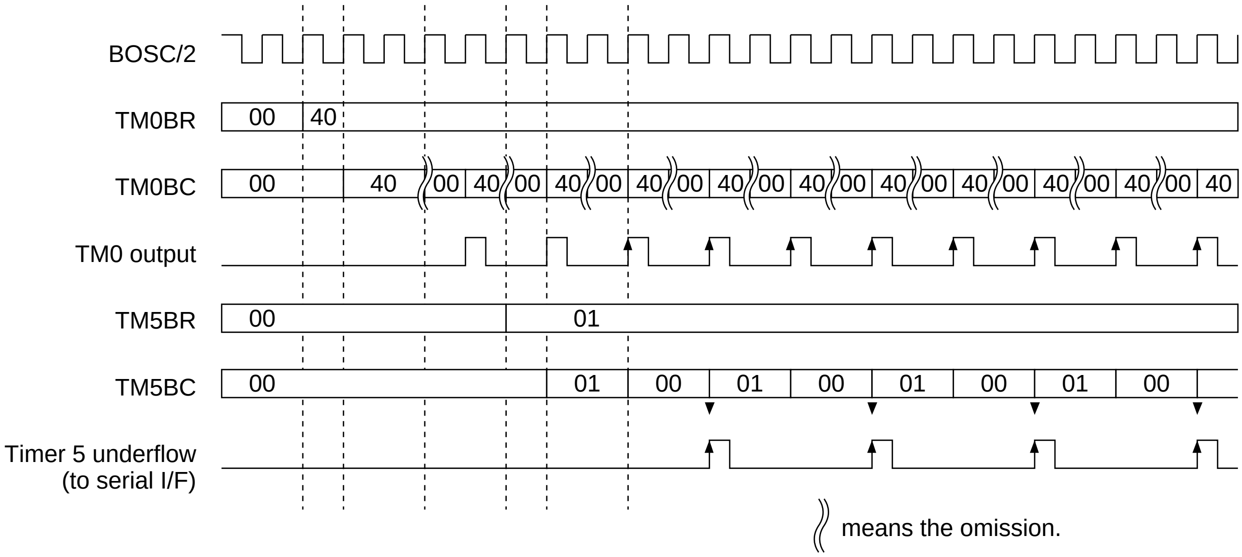


Figure 5-2-6 Serial Clock Timing

5-2-4 I²C Transmission

This section describes the I²C transmission using the serial interface 3. Master transmission is operated using SDA3 and SCL3 pins.

■ Initial Setting

(1) Set the SDA and SCL pins to open drain with the port 8 mode control registers (P8MMD, P8LMD).

P8MMD: x'00FFFD'

7	6	5	4	3	2	1	0
P8 MMD7	P8 MMD6	P8 MMD5	P8 MMD4	P8 MMD3	P8 MMD2	P8 MMD1	P8 MMD0
0	0	0	1	0	0	0	0

P8LMD: x'00FFFC'

7	6	5	4	3	2	1	0
-	-	-	P8 LMD4	P8 LMD3	P8 LMD2	P8 LMD1	P8 LMD0
0	0	0	1	0	1	0	0

(2) Set the serial 3 control register (SC3CTR). Select 8-bit character length, I²C protocol, I²C mode on. The parity bit is set to 1 to enable both transmission and reception enable flags, disable the break and set the ACK output to 1.

ACK is set by the parity bits. To output '1' to ACK, select 1 by the parity bits. To output '0' to ACK, select 0 by the parity bits. To output none to ACK, select none by the parity bits.

SC3CTR: x'00FD98'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SC3 TEN	SC3 REN	SC3 BRE	SC3 IIC	SC3 PTL	-	SC3 OD	SC3 ICM	SC3 LN	SC3 PTY2	SC3 PTY1	SC3 PTY0	SC3 SB	-	SC3 S1	SC3 S0
1	1	0	0	1		1	1	1	1	0	1	0		0	1

■ Start Sequence Transmission

(3) Write 1 to the I²C sequence output flag (SC3IIC) of the SC3CTR register. This sets the SDA3 pin output to low. When the start sequence occurs correctly, the I²C detection flag (SC3IST) of the serial 3 status register (SC3STR) becomes 1. The arbitration lost detection cannot be performed even though the start sequence exists.

Enabling transmission detects the start sequence.

■ Data Transmission 1

(4) Load the data to the serial 3 transmit/receive buffer (SC3TRB). This allows the data to output. The SDA3 pin output changes with a 1/8 cycle delay of the falling edge of the SCL3 pin output.

(5) After transmission ends, SDA3 pin output and SCL3 pin output stay low.

Verify that transmission ends by an interrupt (either a serial 3 transmission end interrupt or a serial 3 reception end interrupt) or polling the received data flag of the serial 3 status register. Polling the reception busy flag is not allowed during I^{2}C mode.

(6) Read the dummy data of the serial 3 transmit/receive buffer (SC3TRB) after transmission ends.

(7) Verify that a parity error occurs by reading the serial 3 status register (SC3STR). When a parity error occurs, this means the slave responds normally. When a parity error does not occur, this means the slave does not respond. (This step is unnecessary for the system without ACK.)

■ **Data Transmission 2**

(8) Repeat steps (4) to (7) if the data is transmitted continuously.

■ **Stop Sequence**

(9) Write 0 to the SC3IIC flag of the SC3CTR register to end the data transmission. Do not write during transmission.

(10) Set the SCL3 pin output to high as soon as the SC3IIC flag is written. One cycle later, set the SDA3 pin output to high to start the stop sequence transmission. The SC3ISP flag of the SC3STR register becomes 1. (Reception must be enabled to detect the stop sequence.) Clear the SC3IST and SC3ISP flags of the SC3STR register by writing to or reading from the SC3TRB register.

(11) Set the SC3REN flag to disable once immediately after the stop sequence occurs.

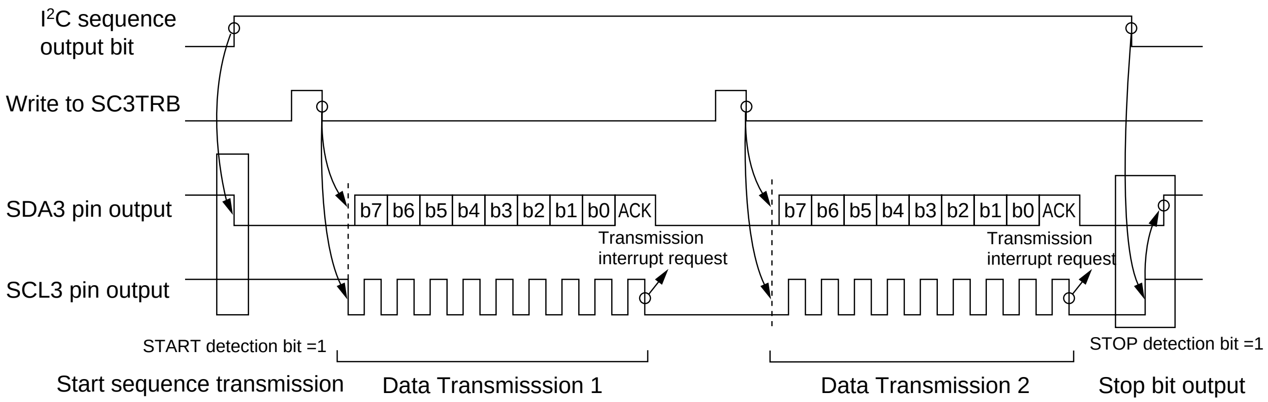


Figure 5-2-7 Master Transmission Timing (With ACK)

5-2-5 I²C Reception

This section describes the I²C reception using the serial interface 3. Master reception is operated using SDA3 and SCL3 pins.

To enter the master reception mode, the first 1 byte must be transmitted during master transmission. Therefore, master reception is performed during the interrupt service routine which runs after the data has been transferred. Please refer to "5-2-4 I²C Transmission" for master transmission.

■ Initial Setting

(1) Enable the reception enable flag (SC3REN) of the serial 3 control register (SC3CTR) during the serial transmission end interrupt service routine.

This step is not required when reception is enabled by the initial setting.

SC3CTR: x'00FD98'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SC3 TEN	SC3 REN	SC3 BRE	SC3 IIC	SC3 PTL	-	SC3 OD	SC3 ICM	SC3 LN	SC3 PTY2	SC3 PTY1	SC3 PTY0	SC3 SB	-	SC3 S1	SC3 S0
1	1	0	0	1		1	1	1	1	0	1	0		0	1

This step can be omitted if it is the same setting in transmission.

ACK is set by the parity bits. To output '1' to ACK, select 1 by the parity bits. To output '0' to ACK, select 0 by the parity bits. To output none to ACK, select none by the parity bits.

■ Data Reception

(2) Load the dummy data x'FF' to the serial 3 transmit/receive buffer (SC3TRB). This starts master reception by setting SDA3 pin output to high.

Write the dummy data x'FF' always.

(3) Retrieve the data by reading the SC3TRB register during the serial reception interrupt service routine. (A serial transmission end interrupt can be served as a serial reception end interrupt.)

Verify that reception ends by an interrupt (either a serial 3 transmission end interrupt or a serial 3 reception end interrupt) or polling the received data flag of the serial 3 status register. Polling the reception busy flag is not allowed during I²C mode.

(4) Load the dummy data x'FF' to the SC3TRB register if the next data is received continuously.

■ Stop Sequence

(5) Write 0 to the SC3IIC flag of the SC3CTR register to start the stop sequence.

(6) The stop sequence output makes the data reception in progress. After the stop sequence is output, disable the reception enable flag and initialize the reception.

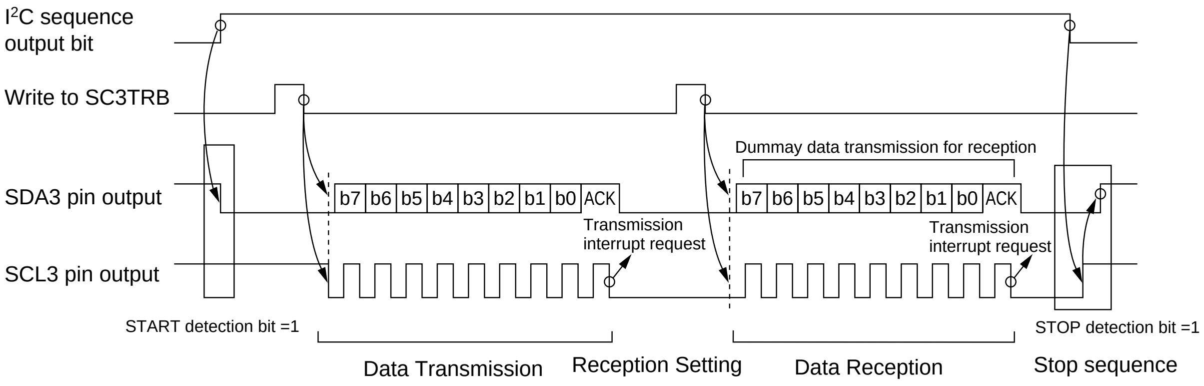


Figure 5-2-8 Master Reception Timing

Chapter 6 Analog Interface

6

6-1 Summary of A/D Converter

6-1-1 Overview

The MN102H55D/55G/F55G contains a 10-bit charge redistribution A/D converter which processes up to 8 channels. Using the clock selection bits, the clock source for A/D converter is selected to BOSC/2, BOSC/4, BOSC/8 or BOSC/16. When BOSC is 30 MHz, the clock source must be set to BOSC/8 (conversion time = 3.73 µs) or higher.

The voltage between Vref+ and Vref- must be input to each analog input pin. Set the voltages of Vref+ pin and Vref- pin as follows:

$$V_{SS} \leq Vref- < Vref+ \leq V_{DD}$$

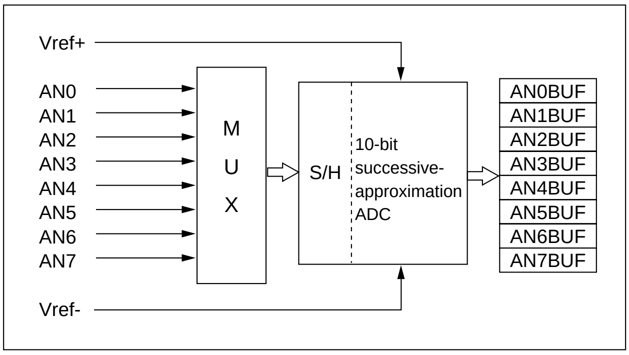


Figure 6-1-1 Analog Interface Configuration

■ **Notices When Using A/D Converter**

(1) Set the impedance of the analog signal for A/D conversion to 8 kΩ or less.

(2) If the impedance of the analog signal cannot be set to 8 kΩ or less, connect the A/D input pin to the condenser of 2000 pF or more to control the voltage change of the A/D input pin.

(3) To prevent the power potential fluctuation, do not change the chip output level from high level to low level or vice verse, or do not switch the peripheral load circuit on/off during A/D conversion.

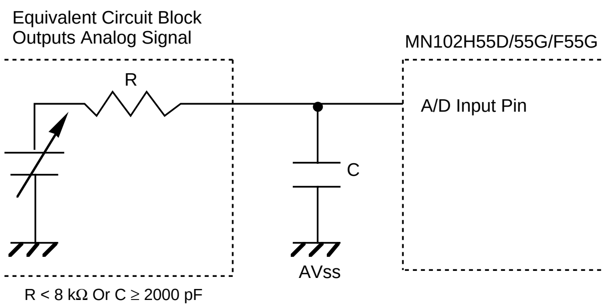


Table 6-1-1 A/D Converter Functions

Feature	Description
Sample and Hold	Built-in
Conversion Resolution	10-bit The A/D converter converts the voltage between Vref+ and Vref- divided into 1024, and this converted result is stored in ANnBUF.
Conversion Time	2.83 µs or more per channel, 3.73 µs per channel with a 30-MHz external oscillator
Clock Source	Internal Clock BOSC divided by 2, 4, 8, 16
Operating Mode	30 operating modes: Single conversion of channel 0 to n (n=1 to 7) Single conversion of channel m (m=0 to 7) Continous conversion of channel 0 to n (n=1 to 7) Continous conversion of channel m (m=0 to 7)
Converstion Start	Timer 3 underflow or register setting
Interrupts	An interrupt occurs each time the conversion sequence ends

■ Selecting the A/D Converter Clock Source

The A/D converter clock source is selected to BOSC/2, BOSC/4, BOSC/8 or BOSC/16 as the A/D conversion time is 2.8 µs or more at 10-bit resolution and 2.4 µs or more at 8-bit resolution.

Calculate the A/D conversion time as follows:

Conversion time (s) (10-bit resolution) = [14 × BOSC cycle/Clock Source]/ch

Conversion time (s) (8-bit resolution) = [12 × BOSC cycle/Clock Source]/ch

For example, when the A/D converter clock source is selected to BOSC/8, the conversion time is BOSC×112 cycles (10-bit resolution). Figure 6-1-2 shows the A/D Converter timing.

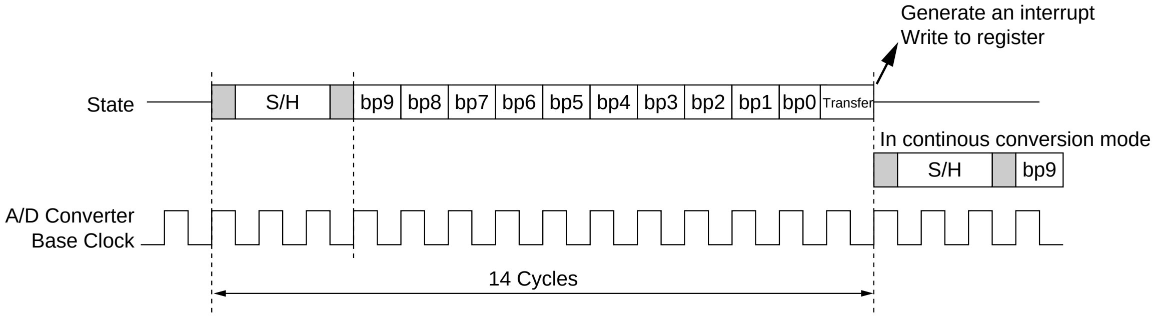


Figure 6-1-2 A/D Converter Timing

Therefore, select the A/D converter clock source as follows:

[Clock Source ≤ 5 MHz/BOSC frequency]

For example, select BOSC/8 or BOSC/16 with a 30-MHz external oscillator since Clock Source ≤ 5 MHz/30 MHz.

■ Single Channel/Single Conversion Timing

When the operating mode selection bits (ANMD[1:0]) are set to single channel/single conversion, the A/D converter converts one AN input signal once. An interrupt occurs when the conversion ends. The number of channel to be converted is set to the channel selection bits (AN1CH[2:0]). (ANNCH[2:0] are ignored.)

When the software starts the conversion, write 0 and 1 to the timer conversion start flag (ANTC) and the conversion start/execution flag (ANEN) of the A/D converter control register (ANCTR) respectively. When ANTC=1, the ANEN flag becomes 1 at timer 3 underflow. The ANEN flag remains 1 during the conversion and clears 0 when the conversion ends.

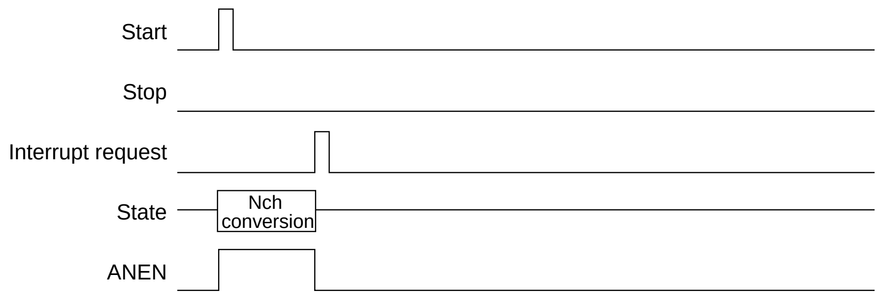


Figure 6-1-3 Single Channel/Single Conversion Timing

■ Multiple Channels/Single Conversion Timing

When the operating mode selection bits (ANMD[1:0]) are set to multiple channels/ single conversion, the A/D converter converts consecutive AN input signals once. An interrupt occurs when the conversion sequence ends. The channel selection bits (AN1CH[2:0]) are set to channel 0 and the number of the final channel to be converted is set to ANNCH[2:0]. The conversion always starts with channel 0.

When the software starts the conversion, write 0 and 1 to the timer conversion start flag (ANTC) and the conversion start/execution flag (ANEN) of the A/D converter control register (ANCTR) respectively. When ANTC=1, the ANEN flag becomes 1 at timer 3 underflow. The ANEN flag remains 1 during the conversion and clears 0 when the conversion ends. AN1CH[2:0] show the number of channel being converted and they clear to 0 when the conversion sequence ends.

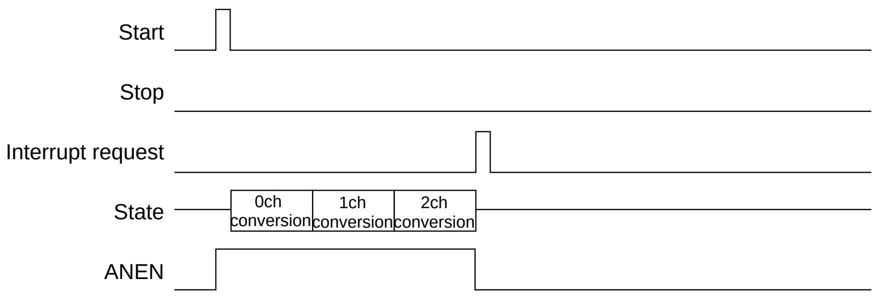


Figure 6-1-4 Multiple Channels/Single Conversion Timing

■ Single Channel/Continuous Conversion Timing

When the operating mode selection bits (ANMD[1:0]) are set to single channel/continuous conversion, the A/D converter converts one AN input signal continuously. An interrupt occurs when the conversion ends. The number of channel to be converted is set to the channel selection bits (AN1CH[2:0]). (ANNCH[2:0] are ignored.)

When the software starts the conversion, write 0 and 1 to the timer conversion start flag (ANTC) and the conversion start/execution flag (ANEN) of the A/D converter control register (ANCTR) respectively. When ANTC=1, the ANEN flag becomes 1 at timer 3 underflow. The ANEN flag remains 1 during the conversion. To end the conversion, write 0 to the ANEN flag.

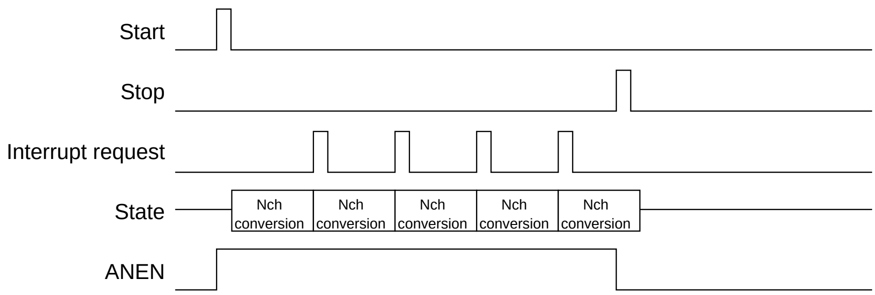


Figure 6-1-5 Single Channel/Continous Conversion Timing

■ Multiple Channels/Continuous Conversion Timing

When the operating mode selection bits (ANMD[1:0]) are set to multiple channels/ continuous conversion, the A/D converter converts multiple, consecutive AN input signals continuously. An interrupt occurs when the conversion sequence ends. The channel selection bits (AN1CH[2:0]) are set to channel 0 and the number of the final channel to be converted is set to ANNCH[2:0]. The conversion always starts with channel 0.

When the software starts the conversion, write 0 and 1 to the timer conversion start flag (ANTC) and the conversion start/execution flag (ANEN) of the A/D converter control register (ANCTR) respectively. When ANTC=1, the ANEN flag becomes 1 at timer 3 underflow. The ANEN flag remains 1 during the conversion. To end the conversion, write 0 ti the ANEN flag. AN1CH[2:0] show the number of channel being converted and they clear to 0 when the conversion sequence ends.

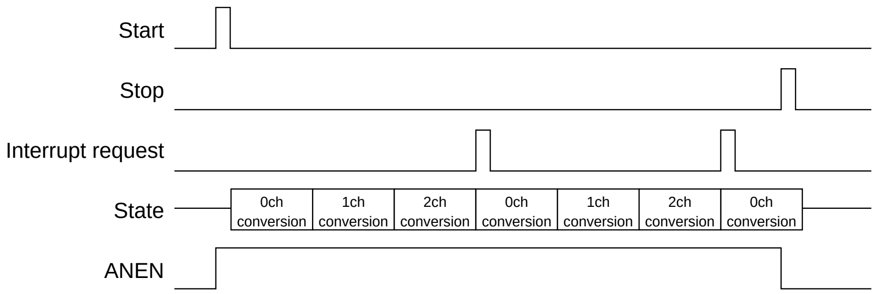


Figure 6-1-6 Multiple Channels/Continous Conversion Timing

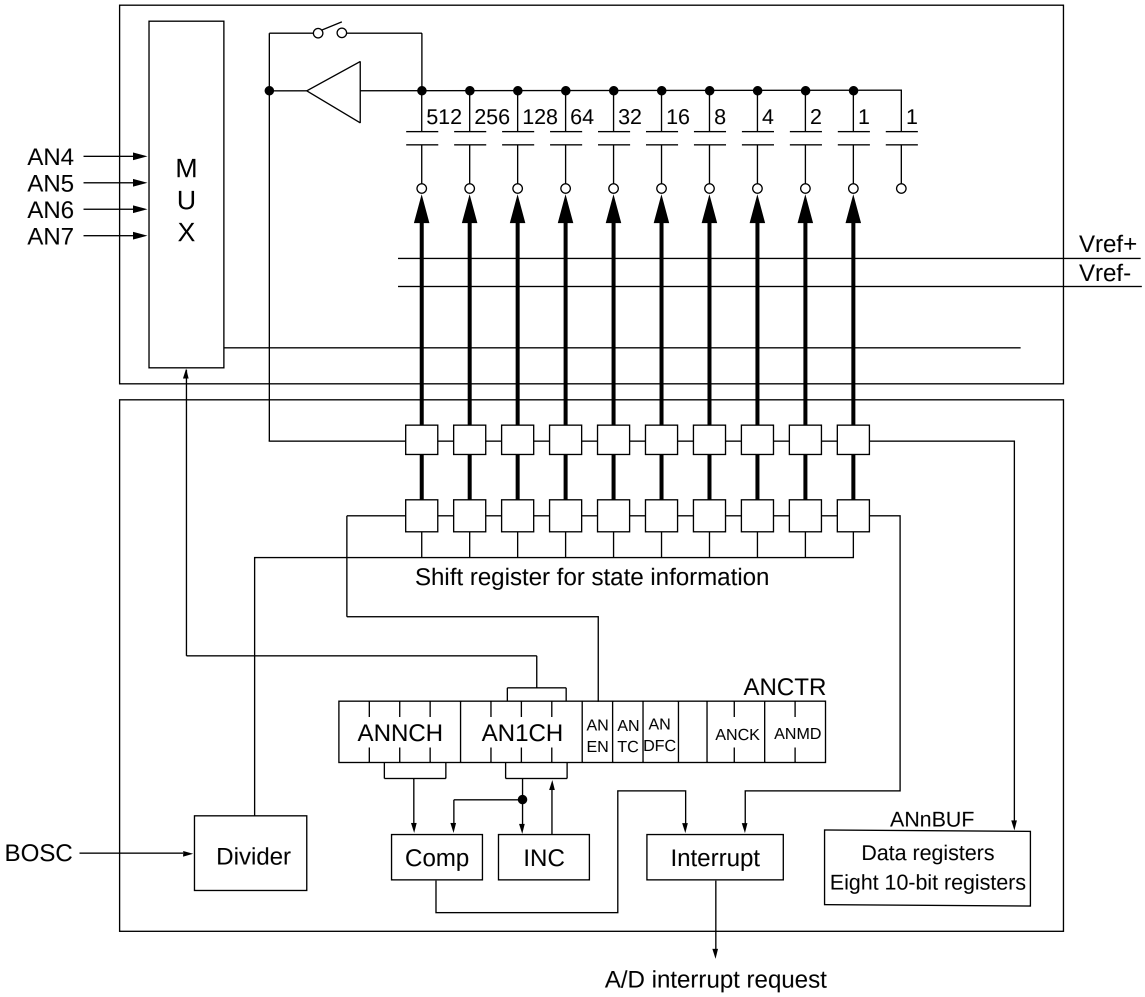


Figure 6-1-7 A/D Converter Block Diagram

6-1-2 Control Registers

The A/D converter contains the A/D converter control register (ANCTR) and the A/D conversion data buffers (ANnBUF) corresponded to AN7 pin to AN0 pin.

ANnBUF

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	-	-	-	-	-	-	ANn BUF9	ANn BUF8	ANn BUF7	ANn BUF6	ANn BUF5	ANn BUF4	ANn BUF3	ANn BUF2	ANn BUF1	ANn BUF0	ANnBUF
Reset:	0	0	0	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	

At 8-bit resolution, the ANnBUF[7:0] bits hold the data and the ANnBUF[9:8] bits become 0. At 10-bit resolution, the ANnBUF[9:0] bits hold the data. At reset the data is undefined.

ANCTR: x'00FF00'

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	AN NCH2	AN NCH1	AN NCH0	—	AN 1CH2	AN 1CH1	AN 1CH0	AN EN	AN TC	AN DEC	—	AN CK1	AN CK0	AN MD1	AN MD0
Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ANMD [1:0]	Operating Mode Selection
00	Single channel, single conversion
01	Multiple channels, single conversion
10	Single channel, continuous conversion
11	Multiple channels, continous conversion

ANCK [1:0]	Clock Source Selection
00	BOSC/2
01	BOSC/4
10	BOSC/8
11	BOSC/16

ANDEC	A/D Converter Resolution
0	8-bit
1	10-bit

ANTC	Conversion Start at Timer 3 Underflow
0	Disable
1	Enable

ANEN	Conversion Start/Execution Flag
0	No conversion
1	Conversion in progress

AN1CH [2:0]	Channel Selection for Single Channel Conversion
000	Convert AN0
001	Convert AN1
010	Convert AN2
011	Convert AN3
100	Convert AN4
101	Convert AN5
110	Convert AN6
111	Convert AN7

ANNCH [2:0]	Channel Selection for Multiple Channel Conversion
000	Convert AN0
001	Convert from AN0 to AN1
010	Convert from AN0 to AN2
011	Convert from AN0 to AN3
100	Convert from AN0 to AN4
101	Convert from AN0 to AN5
110	Convert from AN0 to AN6
111	Convert from AN0 to AN7

Table 6-1-2 List of A/D Converter Control Registers

Register	Address	R/W	Function
ANCTR	x'00FF00'	R/W	A/D Converter Control Register
AN0BUF	x'00FF08'	R	A/D 0 Conversion Data Buffer
AN1BUF	x'00FF0A'	R	A/D 1 Conversion Data Buffer
AN2BUF	x'00FF0C'	R	A/D 2 Conversion Data Buffer
AN3BUF	x'00FF0E'	R	A/D 3 Conversion Data Buffer
AN4BUF	x'00FF10'	R	A/D 4 Conversion Data Buffer
AN5BUF	x'00FF12'	R	A/D 5 Conversion Data Buffer
AN6BUF	x'00FF14'	R	A/D 6 Conversion Data Buffer
AN7BUF	x'00FF16'	R	A/D 7 Conversion Data Buffer

6-2 A/D Converter Setup Examples

6-2-1 Single Channel A/D Conversion

The AN6 pin inputs an analog voltage (Vref- to Vref+) and obtains the 10-bit A/D conversion results.

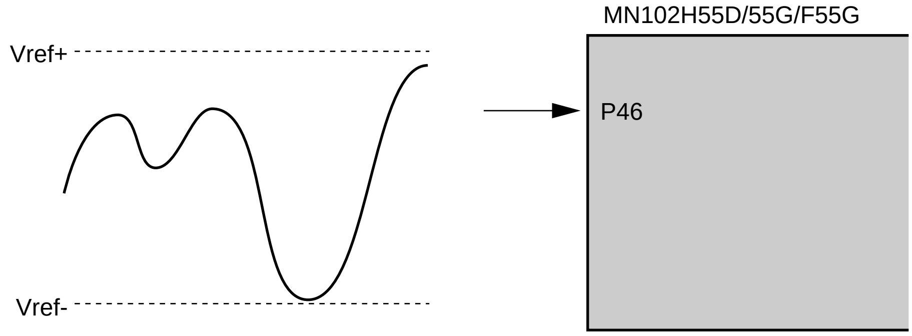


Figure 6-2-1 Analog Voltage Input Example

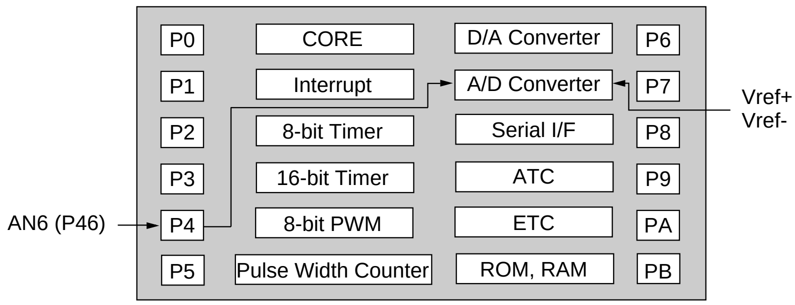


Figure 6-2-2 Single Channel A/D Conversion Block Diagram

■ Port Input and A/D Converter Setup

The P46 direction is always set to input regardless of the P4DIR value.

(1) Set AN6 pin (P46) of the port 4 to AN6 input using the P4HMD register.

(2) Set the operating conditions in the A/D converter control register (ANCTR). Select single channel/single conversion mode by setting ANMD[1:0] to 00, BOSC/8 as the clock source by setting ANCK[1:0] to 10, and 10-bit conversion resolution by setting ANDEC to 1. Set the conversion start/execute flag (ANEN) to 0 and AN1CH[2:0] bits to the number of channel to be converted.

ANCTR: x'00FF00'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	AN NCH2	AN NCH1	AN NCH0	-	AN 1CH2	AN 1CH1	AN 1CH0	AN EN	AN TC	AN DEC	-	AN CK1	AN CK0	AN MD1	AN MD0
0	0	0	0	0	1	1	0	0	0	1	0	1	0	0	0

(3) Set the ANEN flag to 1 to start conversion. Conversion begins on the first rising edge of the A/D converter clock after the ANEN flag is set. The conversion time is 14 cycles of the A/D converter clock (3.73 µs, 3.73 µs to 4.0 µs after the ANEN flag is set).

(4) Wait for conversion to end. The ANEN flag is 1 during the conversion and is cleared to 0 when the conversion is completed. The program waits until the ANEN flag becomes 0.

(5) Read the AN6 conversion data buffer (AN6BUF). The converter divides the voltage between Vref- and Vref+ into 1024, and the conversion result is a value from 0 to 1023.

The CPU can read the result value by generating an interrupt. In this case, the program does not need to wait until the ANEN flag is 0 because an interrupt occurs after the result data is stored in the AN6BUF register.

AN6BUF: x'00FF14'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	AN6 BUF9	AN6 BUF8	AN6 BUF7	AN6 BU6F	AN6 BUF5	AN6 BUF4	AN6 BUF3	AN6 BUF2	AN6 BUF1	AN6 BUF0

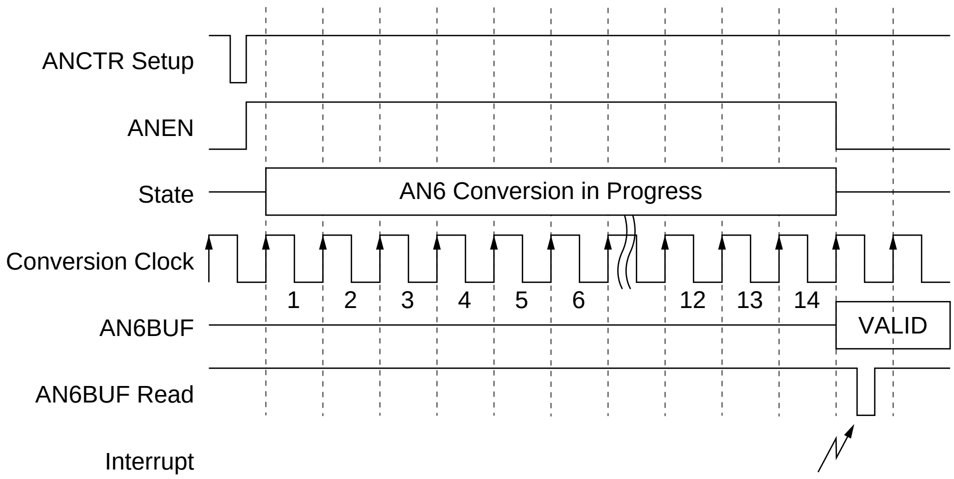


Figure 6-2-3 Single Channel A/D Conversion Timing

6-2-2 Three Channel A/D Conversion

The AN0, AN1 and AN2 pins input analog voltages (Vref+ to Vref-) and the A/D converter converts 8-bit data. The conversion occurs periodically (when timer 3 underflows).

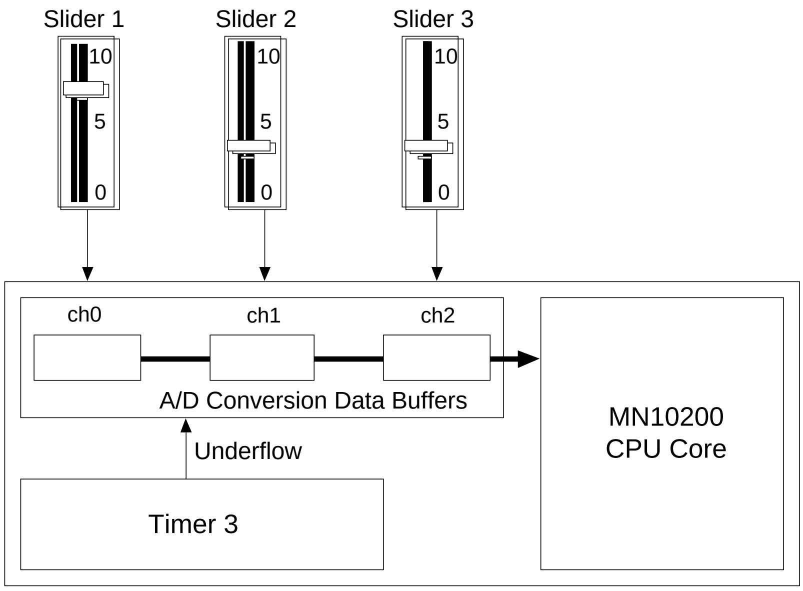


Figure 6-2-4 3-channel A/D Conversion Configuration

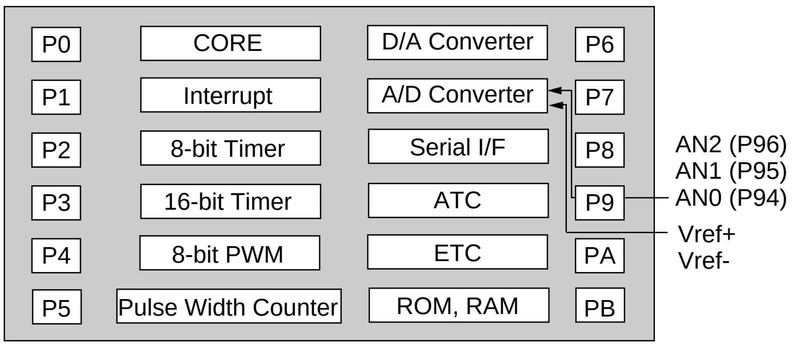


Figure 6-2-5 3-channel A/D Conversion Block Diagram

■ Port Input and A/D Converter Setup

(1) Set AN0, AN1 and AN2 pins (P94, P95 and P96) of the port 8 to input using the P9HMD register.

P9HMD: x'00FFED'

7	6	5	4	3	2	1	0
P9 MD7	P9 MD6	P9 MD5	P9 MD4	P9 MD3	P9 MD2	P9 MD1	P9 MD0
0	0	0	1	1	1	0	0

(2) Set the operating conditions in the A/D converter control register (ANCTR). Select multiple channel/single conversion mode, BOSC/8 as the clock source, and 8-bit conversion resolution. Set the conversion start/execute flag (ANEN) and the ANTC flag to 0 and 1 respectively. Set the AN1CH[2:0] flags to channel 0 and the ANNCH[2:0] flags to the number of the final channel to be converted (2 in this example).

ANCTR: x'00FF00'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	AN NCH2	AN NCH1	AN NCH0	-	AN 1CH2	AN 1CH1	AN 1CH0	AN EN	AN TC	AN DEC	-	AN CK1	AN CK0	AN MD1	AN MD0
0	0	1	0	0	0	0	0	0	1	0	0	1	0	0	1

■ A/D Conversion Interval Setup

(3) Set the divisor for timer 3. To divide BOSC/2 by 256, write 255 to the timer 3 base register (TM3BR). (The valid range is 0 to 255.)

TM3BR: x'00FE13'

7	6	5	4	3	2	1	0
TM3 BR7	TM3 BR6	TM3 BR5	TM3 BR4	TM3 BR3	TM3 BR2	TM3 BR1	TM3 BR0
1	1	1	1	1	1	1	1

(4) Load the value of the TM3BR register to the timer 3 binary counter (TM3BC).

TM3MD: x'00FE23'

7	6	5	4	3	2	1	0
TM3 EN	TM3 LD	-	-	-	-	TM3 S1	TM3 S0
0	1					0	0

Do not change the clock source. Selecting the clock source while controlling the count operation will corrupt the value in the binary counter.

(5) Set TM3LD and TM3EN of the TM3MD register to 0 and 1 respectively. This starts the timer. Counting begins at the start of the next cycle.

When the timer 3 binary counter reaches 0 and loads the value 255 from the timer 3 base register at the next count, a timer 3 underflow interrupt request will be sent to the CPU. The A/D converter converts each AN0 to AN2 a single time at timer 3 underflow.

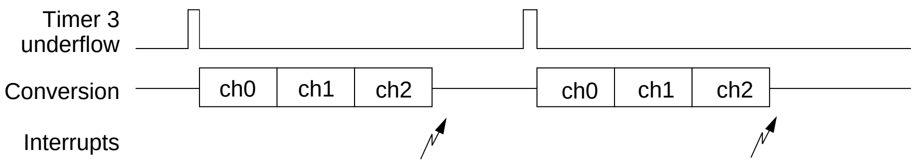


Figure 6-2-6 3-channel A/D Conversion Timing

6-3 Summary of D/A Converter

6-3-1 Overview

The MN102H55D/55G/F55G contains two 8-bit redistribution R-2R D/A converters. Each D/A converter has one output channel and one 8-bit data register. When the D/A converter is unused, turning the ladder resistor off reduces the power current.

DAC[1:0] pins output the voltage (the difference between Vref+ and Vref-) divided by 256 when the data register changes 1 LSB. Set the voltages of Vref+ pin and Vref- pin as follows:

$$V_{SS} \leq Vref- < Vref+ \leq V_{DD}$$

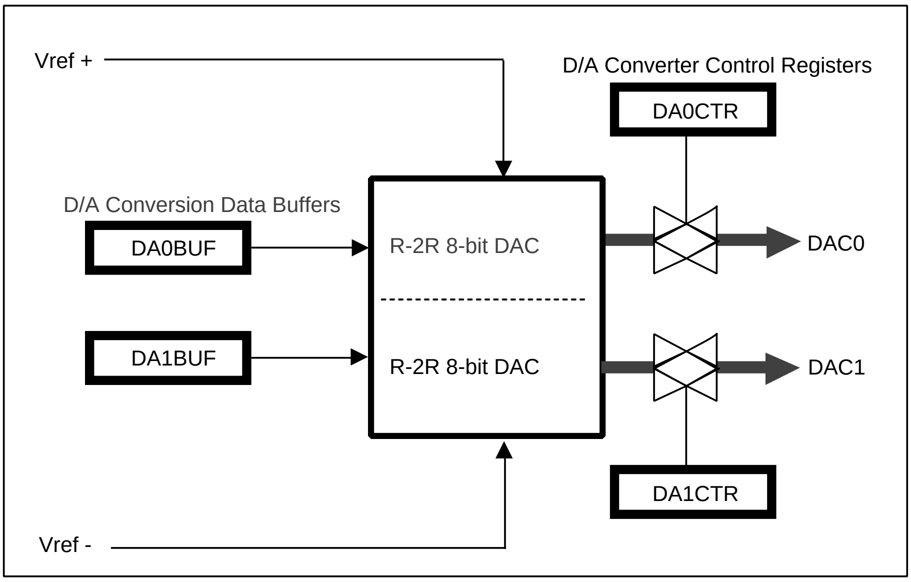


Figure 6-3-1 D/A Converter Configuration

Table 6-3-1 D/A Converter Functions

Feature	R-2R 8-bit D/A Converter
Conversion Resolution	8-bit The voltage corresponding to the value set in DAnBUF (n=0,1) between Vref+ and Vref- is output.
Conversion Time	Maximum of 6.0 µs (External load capacitance 70 pF)

6-3-2 Control Registers

The D/A converter contains the D/A converter control registers (DAnCTR) and the D/A conversion data buffers (DAnBUF) corresponding to DAC1 and DAC0 pins.

DAnCTR

	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	DA0 ON
Reset:	0	0	0	0	0	0	0	0

D/A Conversion Start
0: Disable
1: Enable

DAnBUF

	7	6	5	4	3	2	1	0
	DA0 BUF7	DA0 BUF6	DA0 BUF5	DA0 BUF4	DA0 BUF3	DA0 BUF2	DA0 BUF1	DA0 BUF0
Reset:	0	0	0	0	0	0	0	0

The DAnBUF register stores 8-bit D/A conversion data.

Table 6-3-2 List of D/A Converter Control Registers

Register		Address	R/W	Function
D/A0	DA0CTR	x'00FF40'	R/W	D/A 0 Converter Control Register
	DA0BUF	x'00FF41'	R/W	D/A 0 Conversion Data Buffer
D/A1	DA1CTR	x'00FF42'	R/W	D/A 1 Converter Control Register
	DA1BUF	x'00FF43'	R/W	D/A 1 Conversion Data Buffer

The D/A converter control registers (DAnCTR) set the D/A conversion operating conditions. The D/A conversion data buffers (DAnBUF) input and store the conversion data for channels 1 and 0 (DAC1 and DAC0 pins).

6-4 D/A Converter Setup Examples

6-4-1 D/A Conversion Using DA0 Channel

This section describes D/A conversion circuit operation. The D/A converter setup procedure is the same for DA0 channel and DA1 channel. The conversion data is set in the D/A conversion data buffer and the DAC0 pin (P80) outputs the analog voltage. 1 LSB data set in the D/A conversion data buffer corresponds to the difference between Vref+ and Vref- divided by 256.

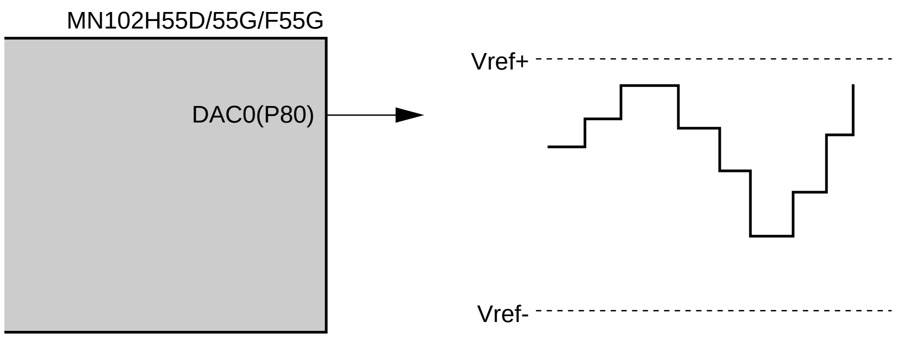


Figure 6-4-1 Analog Voltage Output Example

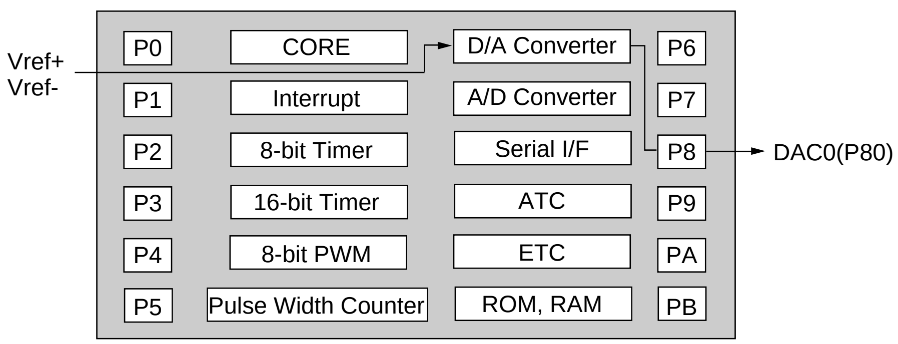


Figure 6-4-2 D/A Conversion Block Diagram Using DA0 Channel

■ Port Output and D/A Converter Setup

(1) Set D/A 0 conversion to start in the D/A 0 converter control register (DA0CTR). This applies the voltage to the ladder resistor for the D/A 0 conversion circuit and increases the power current. To reduce the power current, this setting is not allowed when the D/A conversion is performed.

DA0CTR: x'00FF40'

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	DA0 ON
							1

(2) Set the data to be converted to the D/A 0 conversion data buffer (DA0BUF).

DA0BUF: x'00FF41'

7	6	5	4	3	2	1	0
DA0 BUF7	DA0 BUF6	DA0 BUF5	DA0 BUF4	DA0 BUF3	DA0 BUF2	DA0 BUF1	DA0 BUF0

(3) Set DAC0 (P80) of the port 8 to DAC0 output using the P8LMD register.

P8LMD: x'00FFFC'

7	6	5	4	3	2	1	0
-	-	-	P8 LMD4	P8 LMD3	P8 LMD2	P8 LMD1	P8 LMD0
							1

The DAC0 pin outputs the D/A conversion data.

Chapter 7 ATC, ETC
(Data Automatic Transfer Function)

7-1 Summary of ATC

7-1-1 Overview

The MN102H55D/55G/F55G contains an automatic transfer control (ATC). The ATC has four channels to transfer the data between the memory spaces. The time required from the data transfer request until the data transfer end is the total of the bus acquisition and the data transfer time.

The data transfer time changes depending on the number of waits in the transfer source and the transfer destination. The time required for bus acquisition is a minimum of 1.75 $\times$ internal operating cycle(s) after the ATC receives a data transfer request. For example, if the internal operating cycle is 66.7 ns (with a 30-MHz external oscillator), the time for bus acquisition is 116.725 ns.

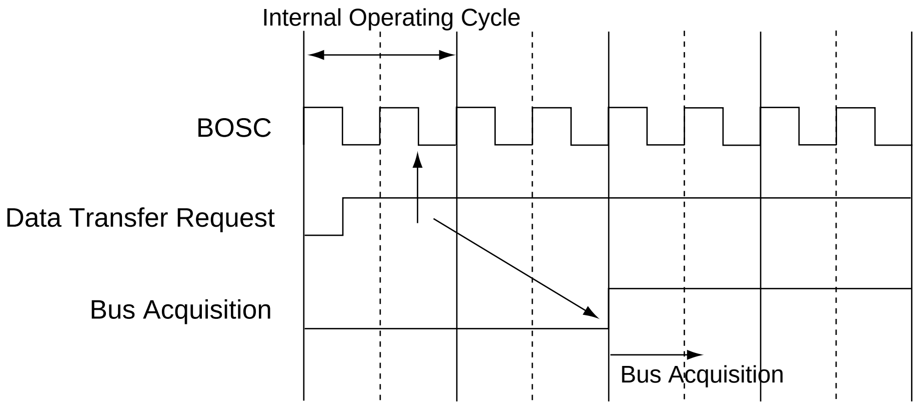


Figure 7-1-1 ATC Bus Acquisition Timing

After bus is acquired, the time required for the data transfer is calculated as follows:

(4+Ws+Wd) $\times$ m $\times$ internal operating cycle(s)

where m: the number of data transfer words

Ws: the number of waits in the source

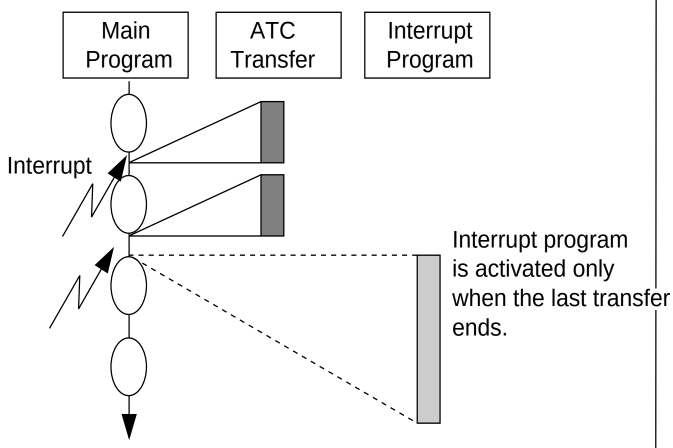
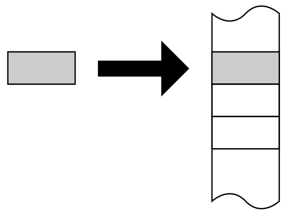
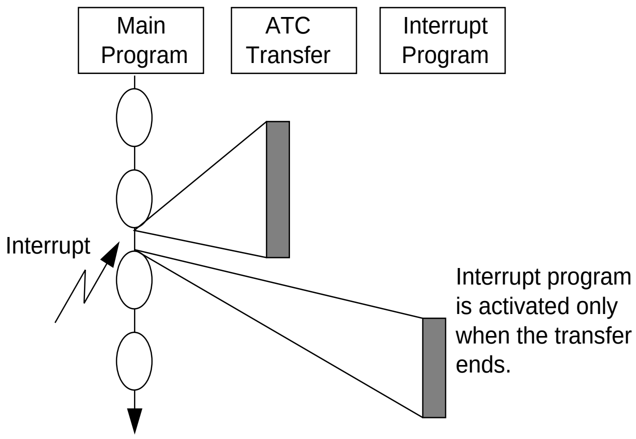
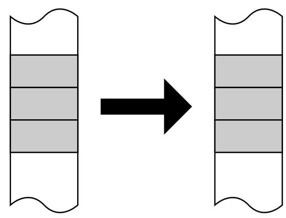
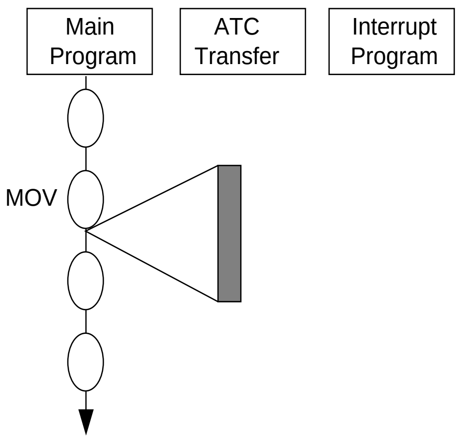
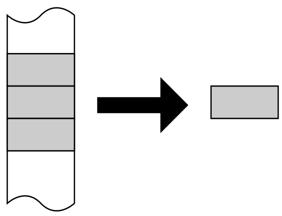
Wd: the number of waits in the destination

After the transfer ends, an ATC transfer end interrupt occurs. ATC does not accept an interrupt except $\overline{\text{NMI}}$ during transfer, but ATC accepts an interrupt after the transfer ends. When $\overline{\text{NMI}}$ occurs during transfer, ATC stops the transfer and executes the interrupt service routine.

The bus acquisition priority is as follows:

ATC0 > ATC1 > ATC2 > ATC3 > CPU

Table 7-1-1 ATC Functions

	Mode	Operation	Memory Operation by ATC Transfer
Serial Reception	One Byte/Word Transfer		
DMA Transfer Using Interrupt	Burst		
Data Transfer to FDC	One Byte/Word Transfer		

Interrupt program activation can be set optionally.

⬯ shows one instruction.

7-1-2 Control Registers

The ATC contains the ATC control registers (ATnCTR) and the ATC transfer word count registers (ATnCNT), the source address pointers (ATnSRC) and the destination address pointers (ATnDST).

ATnCTR

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ATn EN	ATn MD1	ATn MD0	ATn BW	ATn DB8	ATn DI	ATn SB8	ATn SI	—	—	—	—	ATn IQ3	ATn IQ2	ATn IQ1	ATn IQ0
Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ATnIQ[3:0]	ATC Activation Factor Settings
0000	Software initialization
0001	$\overline{\text{DMAREQ0}}$ pin input
0010	External interrupt 0
0011	External interrupt 1
0100	External interrupt 4
0101	Timer 3 underflow interrupt
0110	Timer 7 underflow interrupt
0111	Timer 9 underflow interrupt
1000	Timer 10 capture A interrupt
1001	Timer 11 capture A interrupt
1010	Serial 0 transmission end interrupt
1011	Serial 0 reception end interrupt
1100	Serial 3 transmission end interrupt
1101	Serial 3 reception end interrupt
1110	A/D conversion end interrupt
1111	Key interrupt

Note: This example is ATC0. Please refer to List of Registers for ATC1-ATC3 settings.

ATnSI	Increment Control Flag for Source Address Pointer
0	Fixed
1	Increment by 1 in the byte transfer, by 2 in the word transfer

ATnSB8	Source Bus Width
0	16-bit
1	8-bit (only when 8-bit bus width for external memory is selected)

ATnDI	Increment Control Flag for Destination Address Pointer
0	Fixed
1	Increment by 1 in the byte transfer, by 2 in the word transfer

ATnDB8	Destination Bus Width
0	16-bit
1	8-bit (only when 8-bit bus width for external memory is selected)

ATnBW	Transfer Unit
0	Word (disable when 8-bit bus width for external memory is selected)
1	Byte

ATnMD [1:0]	Transfer Mode
00	One byte/word transfer
01	Burst Transfer
10	Two bytes/words transfer (available in the byte transfer)
11	Reserved

Note: The LSB of address forcibly becomes 0 in the first byte data transfer and the LSB of address forcibly becomes 1 in the second byte data transfer.

ATnEN	Transfer Start/Busy Flag
0	Disable
1	Transfer start/Transfer in progress

Note: Do not activate ATC by an interrupt and write 0 to ATnEN flag by the user program simultaneously. Omitting this procedure causes the CPU to stop. Set the applicable register not to generate an interrupt for ATC activation factor before writing 0 to ATnEN flag.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	-	-	-	-	ATn CNT11	ATn CNT10	ATn CNT9	ATn CNT8	ATn CNT7	ATn CNT6	ATn CNT5	ATn CNT4	ATn CNT3	ATn CNT2	ATn CNT1	ATn CNT0	ATnCNT
Reset:	0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	

The ATnCNT register writes only 16-bit data. Use the MOV instruction to set the data.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	ATn SRC15	ATn SRC14	ATn SRC13	ATn SRC12	ATn SRC11	ATn SRC10	ATn SRC9	ATn SRC8	ATn SRC7	ATn SRC6	ATn SRC5	ATn SRC4	ATn SRC3	ATn SRC2	ATn SRC1	ATn SRC0	ATnSRC
Reset:	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ATn SRC23	ATn SRC22	ATn SRC21	ATn SRC20	ATn SRC19	ATn SRC18	ATn SRC17	ATn SRC16
Reset:	0	0	0	0	0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined

The ATnSRC register and the ATnDST register write only 24-bit data or 16-bit data. Use the MOV instruction or the MOVX instruction to set the data.

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	
	ATn DST15	ATn DST14	ATn DST13	ATn DST12	ATn DST11	ATn DST10	ATn DST9	ATn DST8	ATn DST7	ATn DST6	ATn DST5	ATn DST4	ATn DST3	ATn DST2	ATn DST1	ATn DST0	ATnDST
Reset:	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ATn DST23	ATn DST22	ATn DST21	ATn DST20	ATn DST19	ATn DST18	ATn DST17	ATn DST16
Reset:	0	0	0	0	0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined

Table 7-1-2 List of ATC Control Registers

Register		Address	R/W	Function
ATC0	AT0CTR	x'00FD00'	R/W	ATC 0 Control Register
	AT0CNT	x'00FD02'	R/W	ATC 0 Transfer Word Count Register
	AT0SRC	x'00FD04'	R/W	ATC 0 Source Address Pointer
	AT0DST	x'00FD08'	R/W	ATC 0 Destination Address Pointer
ATC1	AT1CTR	x'00FD10'	R/W	ATC 1 Control Register
	AT1CNT	x'00FD12'	R/W	ATC 1 Transfer Word Count Register
	AT1SRC	x'00FD14'	R/W	ATC 1 Source Address Pointer
	AT1DST	x'00FD18'	R/W	ATC 1 Destination Address Pointer
ATC2	AT2CTR	x'00FD20'	R/W	ATC 2 Control Register
	AT2CNT	x'00FD22'	R/W	ATC 2 Transfer Word Count Register
	AT2SRC	x'00FD24'	R/W	ATC 2 Source Address Pointer
	AT2DST	x'00FD28'	R/W	ATC 2 Destination Address Pointer
ATC3	AT3CTR	x'00FD30'	R/W	ATC 3 Control Register
	AT3CNT	x'00FD32'	R/W	ATC 3 Transfer Word Count Register
	AT3SRC	x'00FD34'	R/W	ATC 3 Source Address Pointer
	AT3DST	x'00FD38'	R/W	ATC 3 Destination Address Pointer

7-2 ATC Setup Examples

7-2-1 Serial Reception

The serial interface 0 receives the 5-byte data. After the serial reception is completed, ATC reads the data using the serial reception buffer and writes the data on the memory. After that, ATC generates an interrupt and starts software processing.

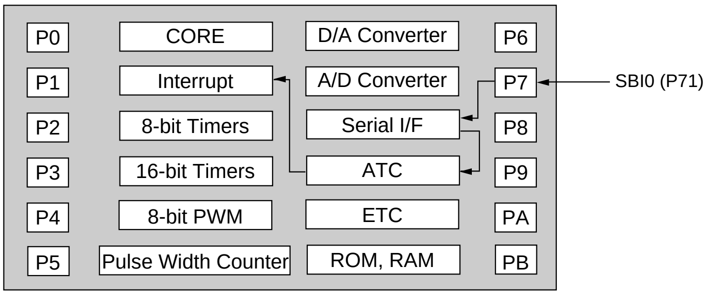


Figure 7-2-1 ATC Serial Reception Block Diagram

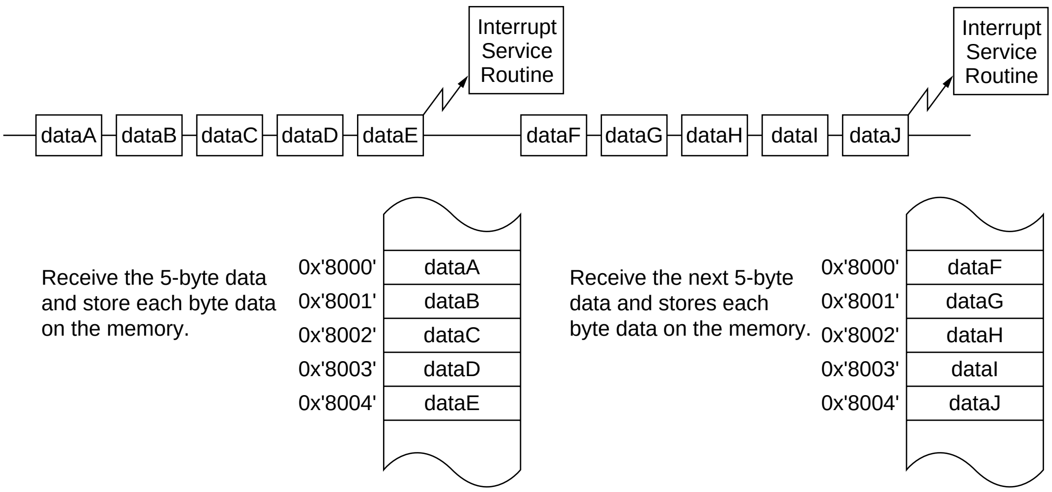


Figure 7-2-2 Serial Reception Sequence

■ ATC Setup

(1) Set the address x'00FD82' of the serial 0 reception buffer to the ATC0 source address pointer (AT0SRC).

AT0SRC: x'00FD04'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AT0 SRC15	AT0 SRC14	AT0 SRC13	AT0 SRC12	AT0 SRC11	AT0 SRC10	AT0 SRC9	AT0 SRC8	AT0 SRC7	AT0 SRC6	AT0 SRC5	AT0 SRC4	AT0 SRC3	AT0 SRC2	AT0 SRC1	AT0 SRC0
1	1	1	1	1	1	0	1	1	0	0	0	0	0	1	0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	AT0 SRC23	AT0 SRC22	AT0 SRC21	AT0 SRC20	AT0 SRC19	AT0 SRC18	AT0 SRC17	AT0 SRC16
								0	0	0	0	0	0	0	0

(2) Secure the space for the 5-byte serial 0 reception data. Set the first address of the secured space to the ATC0 destination address pointer (AT0DST).

The space for 5 bytes is from x'008000' to x'008004'.

AT0DST: x'00FD08'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AT0 DST15	AT0 DST14	AT0 DST13	AT0 DST12	AT0 DST11	AT0 DST10	AT0 DST9	AT0 DST8	AT0 DST7	AT0 DST6	AT0 DST5	AT0 DST4	AT0 DST3	AT0 DST2	AT0 DST1	AT0 DST0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	AT0 DST23	AT0 DST22	AT0 DST21	AT0 DST20	AT0 DST19	AT0 DST18	AT0 DST17	AT0 DST16
								0	0	0	0	0	0	0	0

(3) Set the bytes to be transferred automatically. In this example, 5-byte data is transferred so that the value '4' subtracting 5 by 1 is set to the ATC0 transfer word count register (AT0CNT).

AT0CNT: x'00FD02'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	AT0 CNT11	AT0 CNT10	AT0 CNT9	AT0 CNT8	AT0 CNT7	AT0 CNT6	AT0 CNT5	AT0 CNT4	AT0 CNT3	AT0 CNT2	AT0 CNT1	AT0 CNT0
0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0

(4) Set the ATC0 control register (AT0CTR). Select a serial 0 reception end interrupt. Set the source pointer to be fixed and the destination pointer to increment by 1. Select one byte unit and one byte/word transfer as the transfer mode. Set the transfer start/busy flag to disable. Select 16-bit as both source bus width and destination bus width.

Select 16-bit source bus width and 16-bit destination bus width. Select 8-bit source bus width -bit source bus width and 8-bit destination bus width only when 8-bit bus width for the external memory space is selected.

AT0CTR: x'00FD00'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AT0 EN	AT0 MD1	AT0 MD0	AT0 BW	AT0 DB8	AT0 DI	AT0 SB8	AT0 SI	-	-	-	-	AT0 IQ3	AT0 IQ2	AT0 IQ1	AT0 IQ0
0	0	0	1	0	1	0	0	0	0	0	0	1	0	1	1

(5) Enable an ATC0 transfer end interrupt.

The interrupt level is 5 in this example.

AT0ICH: x'00FCA9'

7	6	5	4	3	2	1	0
-	AT0 LV2	AT0 LV1	AT0 LV0	-	-	-	AT0 IE
	1	0	1				1

In this example, an error cannot be detected during the transfer. When an error is needed to be detected, set the AT0BW flag of the AT0CTR register to 1 to enable the word transfer. This allows to transfer the data between the SC0TRB register and the SC0STR register. The 5-word (10-byte) memory space is required. Checking the contents of the SC0STR register transferred to the memory during the interrupt service routine indicates each reception status.

■ Serial Setup

(6) Disable a serial 0 reception end interrupt. (If an interrupt is enabled, the serial 0 reception end interrupt is processed after ATC one-byte transfer ends.)

SC0RICH: x'00FC92'

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	SC0R IE
0	0	0	0	0	0	0	0

(7) Select serial reception mode. Refer to "Serial Interface Setup Examples" for details.

■ ATC Reset

(8) Process the 5-byte serial 0 reception data. Each ATC register value is set as follows:

AT0CTR: x'00FD00'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AT0 EN	AT0 MD1	AT0 MD0	AT0 BW	AT0 DB8	AT0 DI	AT0 SB8	AT0 SI	-	-	-	-	AT0 IQ3	AT0 IQ2	AT0 IQ1	AT0 IQ0
0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0
Busy flag indication	Hold	Hold	Hold	Hold	Hold	Hold	Hold					Reset after 5-byte data transfer			

AT0CNT: x'0FFF' (This value is always set regardless of the bytes to be transferred.)
AT0SRC: x'00FD82' (The last value is stored.)
AT0DST: x'008005' (The result incremented by 1 is set after the last transfer is completed.)

If this setting is omitted, the 4096-byte data is transferred because the AT0CNT value is x'0FFF'.

(9) Secure the space for the 5-byte serial 0 reception data. Reset the first address of the secured space to the ATC0 destination address pointer (AT0DST).

(10) Set the bytes to be transferred automatically. In this example, 5-byte data is transferred so that the value '4' subtracting 5 by 1 is set to the ATC0 transfer word count register (AT0CNT).

AT0CNT: x'00FD02'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	AT0 CNT11	AT0 CNT10	AT0 CNT9	AT0 CNT8	AT0 CNT7	AT0 CNT6	AT0 CNT5	AT0 CNT4	AT0 CNT3	AT0 CNT2	AT0 CNT1	AT0 CNT0
				0	0	0	0	0	0	0	0	0	1	0	0

(11) Set the ATC0 control register (AT0CTR). Reselect a serial 0 reception end interrupt. Select the conditions as set in procedure (4).

(12) Verify that a serial 0 reception end interrupt does not occur. If the serial 0 reception end interrupt occurs, avoid the interrupt by setting the AT0EN flag of the AT0CTR register to 1 (start the first byte data transfer by software). The data after the second byte is transferred automatically with the serial reception end interrupt.

7-3 Summary of ETC

7-3-1 Overview

The MN102H55D/55G/F55G contains an external transfer control (ETC). The ETC has two channels to transfer the data between the external memory and the external device. The data transfer request occurs when $\overline{\text{DMAREQ}}$[1:0] become low. $\overline{\text{DMAACK}}$[1:0] become low when the ETC accepts the data transfer request. The time required from the data transfer request until the data transfer end is the total of the bus acquisition and the data transfer time.

The data transfer time changes depending on the number of waits in the transfer source and the transfer destination. The time required for bus acquisition is a minimum of 1.75 × internal operating cycle(s) after the ETC receives a data transfer request. For example, if the internal operating cycle is 66.7 ns (with a 30-MHz external oscillator), the time for bus acquisition is 116.725 ns.

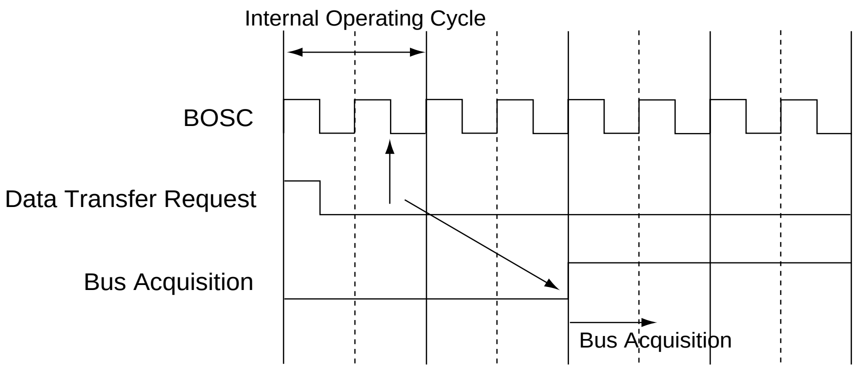


Figure 7-3-1 ETC Bus Acquisition Timing

After bus is acquired, the time required for the data transfer is calculated as follows:

(4+Ws+Wd) × m × internal operating cycle(s)

where m: the number of data transfer words

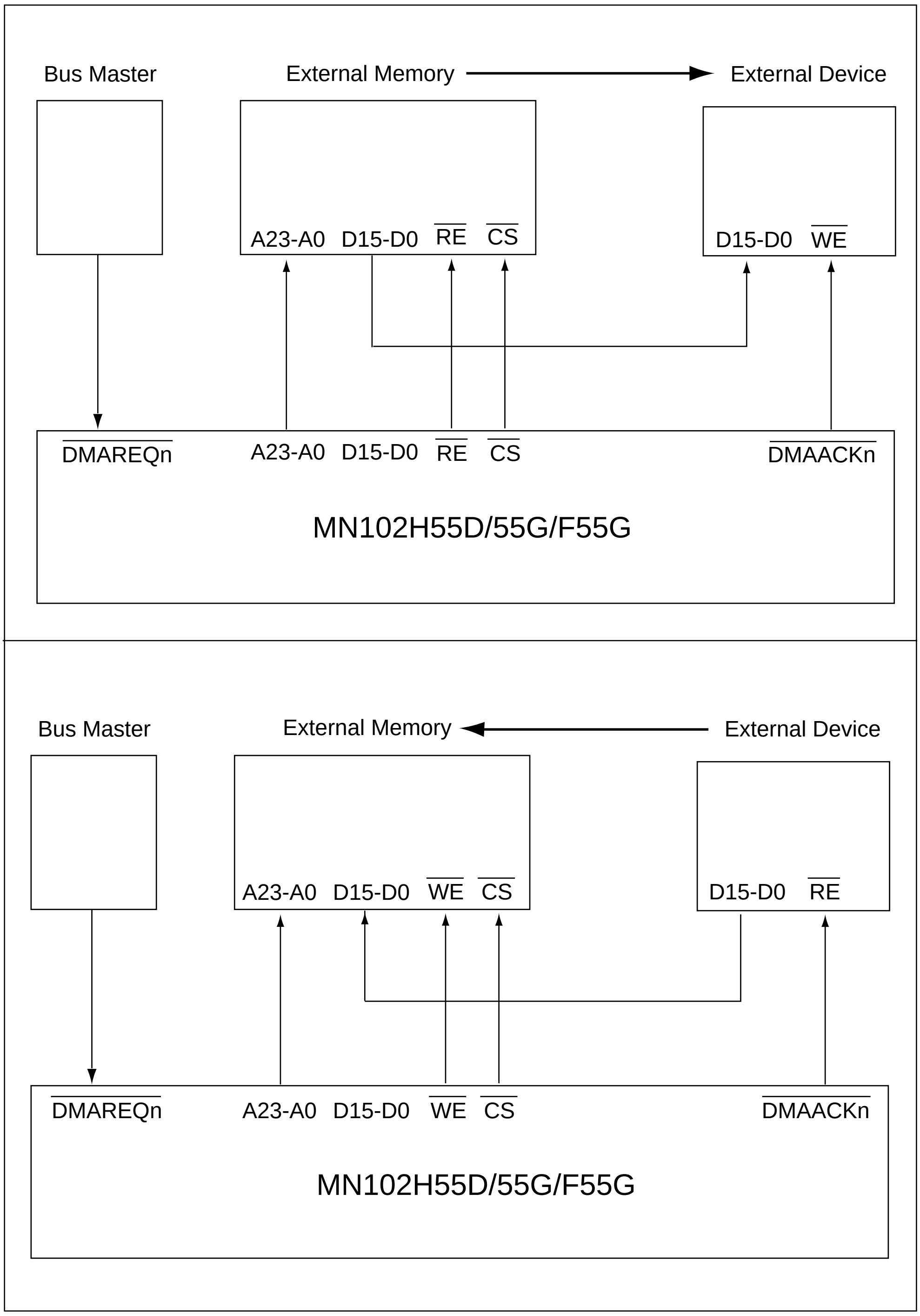
Ws: the number of waits in the source

Wd: the number of waits in the destination

After the transfer ends, an ETC transfer end interrupt occurs. ETC does not accept an interrupt except $\overline{\text{NMI}}$ during transfer, but the ETC accepts an interrupt after the transfer ends. When $\overline{\text{NMI}}$ occurs during transfer, the ETC stops the transfer and executes the interrupt service routine.

The bus acquisition priority is ETC0 > ETC1 > CPU

Table 7-3-1 ETC Connection Examples



ETC stops executing the CPU's program and transfers the data automatically between the external memory and the external device when low level is input to $\overline{\text{DMAREQn}}$ pin from bus master. $\overline{\text{DMAACKn}}$ becomes $\overline{\text{RE}}$ or $\overline{\text{WE}}$ signal for the external device. After the transfer ends, ETC restarts executing the program.

External memory is a device (such as SRAM) that has address input pins, data input/output pins, the $\overline{\text{RE}}$ control pin and the $\overline{\text{WE}}$ control pin. The external memory is connected to the chip in processor mode or address expansion mode with either address/data separate mode or address/data shared mode. The external memory has a register to set the number of waits.

External device is a device (such as ASIC) that has data input/output pins, the $\overline{\text{RE}}$ control pin and the $\overline{\text{WE}}$ control pin without using address input pins. The external device needs to output the data when a signal is input to $\overline{\text{RE}}$ and read the data when a signal is input to $\overline{\text{WE}}$. When waits are required for accesses, the number of waits is set using the register in the external memory.

7-3-2 Control Registers

The ETC contains the ETC control registers (ETnCTR) and the ETC transfer word count registers (ETnCNT), the source address pointers (ETnSRC) and the destination address pointers (ETnDST).

ETnCTR

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ETn FLG	—	ETn MD0	ETn BW	ETn DB8	ETn DI	ETn SB8	ETn SI	—	—	—	—	—	—	ETn DIR	ETn EN
Reset:	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

ETnEN	ETC Transfer Enable
0	Disable
1	Enable

ETnDIR	Transfer Direction between External Device and External Memory
0	From External Device to External Memory
1	From External Memory to External Device

ETnSI	Increment Control Flag for Source Address Pointer
0	Fixed
1	Increment by 1 in the byte transfer, by 2 in the word transfer

ETnSB8	Source Bus Width
0	16-bit
1	8-bit (only when 8-bit bus width for external memory is selected)

ETnDI	Increment Control Flag for Destination Address Pointer
0	Fixed
1	Increment by 1 in the byte transfer, by 2 in the word transfer

ETnDB8	Destination Bus Width
0	16-bit
1	8-bit (only when 8-bit bus width for external memory is selected)

ETnBW	Transfer Unit
0	Word
1	Byte

ETnMD0	Transfer Mode
0	One byte/word transfer
1	Burst transfer

ETnFLG	Transfer Start/Busy Flag
0	Disable
1	Transfer start/Transfer in progress

ETnCNT

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	ETn CNT11	ETn CNT10	ETn CNT9	ETn CNT8	ETn CNT7	ETn CNT6	ETn CNT5	ETn CNT4	ETn CNT3	ETn CNT2	ETn CNT1	ETn CNT0
Reset:	0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined

The ETnCNT register writes only 16-bit data. Use the MOV instruction to set the data.

ETnSRC

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ETn SRC15	ETn SRC14	ETn SRC13	ETn SRC12	ETn SRC11	ETn SRC10	ETn SRC9	ETn SRC8	ETn SRC7	ETn SRC6	ETn SRC5	ETn SRC4	ETn SRC3	ETn SRC2	ETn SRC1	ETn SRC0
Reset:	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ETn SRC23	ETn SRC22	ETn SRC21	ETn SRC20	ETn SRC19	ETn SRC18	ETn SRC17	ETn SRC16
Reset:	0	0	0	0	0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined

The ETnSRC register and the ETnDST register write only 24-bit data or 16-bit data. Use the MOV instruction or the MOVX instruction to set the data.

ETnDST

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ETn DST15	ETn DST14	ETn DST13	ETn DST12	ETn DST11	ETn DST10	ETn DST9	ETn DST8	ETn DST7	ETn DST6	ETn DST5	ETn DST4	ETn DST3	ETn DST2	ETn DST1	ETn DST0
Reset:	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ETn DST23	ETn DST22	ETn DST21	ETn DST20	ETn DST19	ETn DST18	ETn DST17	ETn DST16
Reset:	0	0	0	0	0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined

Table 7-3-2 List of ETC Control Registers

Register		Address	R/W	Function
ETC0	ET0CTR	x'00FD40'	R/W	ETC 0 Control Register
	ET0CNT	x'00FD42'	R/W	ETC 0 Transfer Word Count Register
	ET0SRC	x'00FD44'	R/W	ETC 0 Source Address Pointer
	ET0DST	x'00FD48'	R/W	ETC 0 Destination Address Pointer
ETC1	ET1CTR	x'00FD50'	R/W	ETC 1 Control Register
	ET1CNT	x'00FD52'	R/W	ETC 1 Transfer Word Count Register
	ET1SRC	x'00FD54'	R/W	ETC 1 Source Address Pointer
	ET1DST	x'00FD58'	R/W	ETC 1 Destination Address Pointer

7-4 ETC Setup Examples

7-4-1 Transfer from External Memory to External Device

$\overline{\text{DMAREQ0}}$ input from bus master is an activation factor. Each byte-data is transferred from the external memory to the external device.

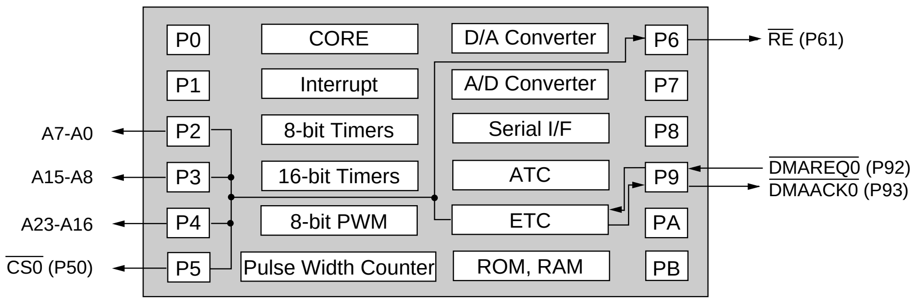


Figure 7-4-1 ETC External Memory → External Device Transfer Block Diagram

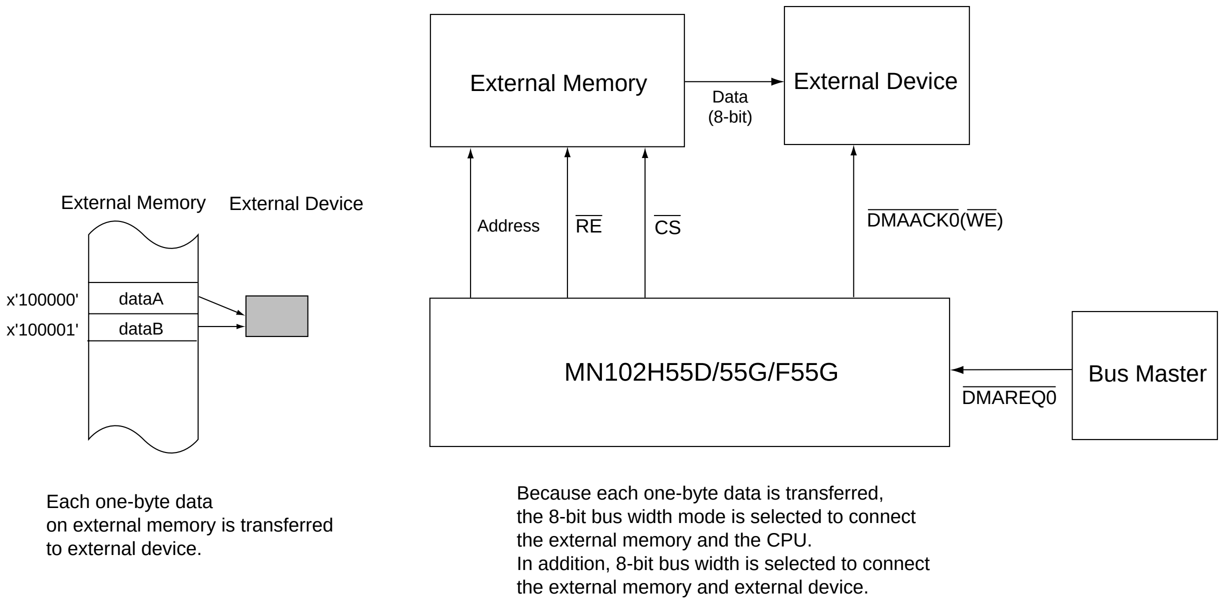


Figure 7-4-2 ETC External Memory → External Device Transfer Connection

■ ETC Setup

(1) Set the source address x'100000' of the external memory to the ETC0 source address pointer (ET0SRC).

ET0SRC: x'00FD44'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ET0 SRC15	ET0 SRC14	ET0 SRC13	ET0 SRC12	ET0 SRC11	ET0 SRC10	ET0 SRC9	ET0 SRC8	ET0 SRC7	ET0 SRC6	ET0 SRC5	ET0 SRC4	ET0 SRC3	ET0 SRC2	ET0 SRC1	ET0 SRC0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	ET0 SRC23	ET0 SRC22	ET0 SRC21	ET0 SRC20	ET0 SRC19	ET0 SRC18	ET0 SRC17	ET0 SRC16
								0	0	0	1	0	0	0	0

(2) Set the bytes to be transferred automatically. In this example, 2-byte data is transferred so that the value '1' subtracting 2 by 1 is set to the ETC0 transfer word count register (ET0CNT).

ET0CNT: x'00FD42'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	ET0 CNT11	ET0 CNT10	ET0 CNT9	ET0 CNT8	ET0 CNT7	ET0 CNT6	ET0 CNT5	ET0 CNT4	ET0 CNT3	ET0 CNT2	ET0 CNT1	ET0 CNT0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

(3) Set the ETC0 control register (ET0CTR). Select burst transfer mode. Select one byte unit and the source pointer to increment by 1. Select the transfer direction is from external memory to external device. Set the transfer start/busy flag to disable.

ET0CTR: x'00FD40'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ET0 FLG	-	ET0 MD0	ET0 BW	ET0 DB8	ET0 DI	ET0 SB8	ET0 SI	-	-	-	-	-	-	ET0 DIR	ET0 EN
0	0	1	1	0	0	1	1	0	0	0	0	0	0	1	1

■ Interrupt Setup

(4) Enable an ETC0 transfer end interrupt.

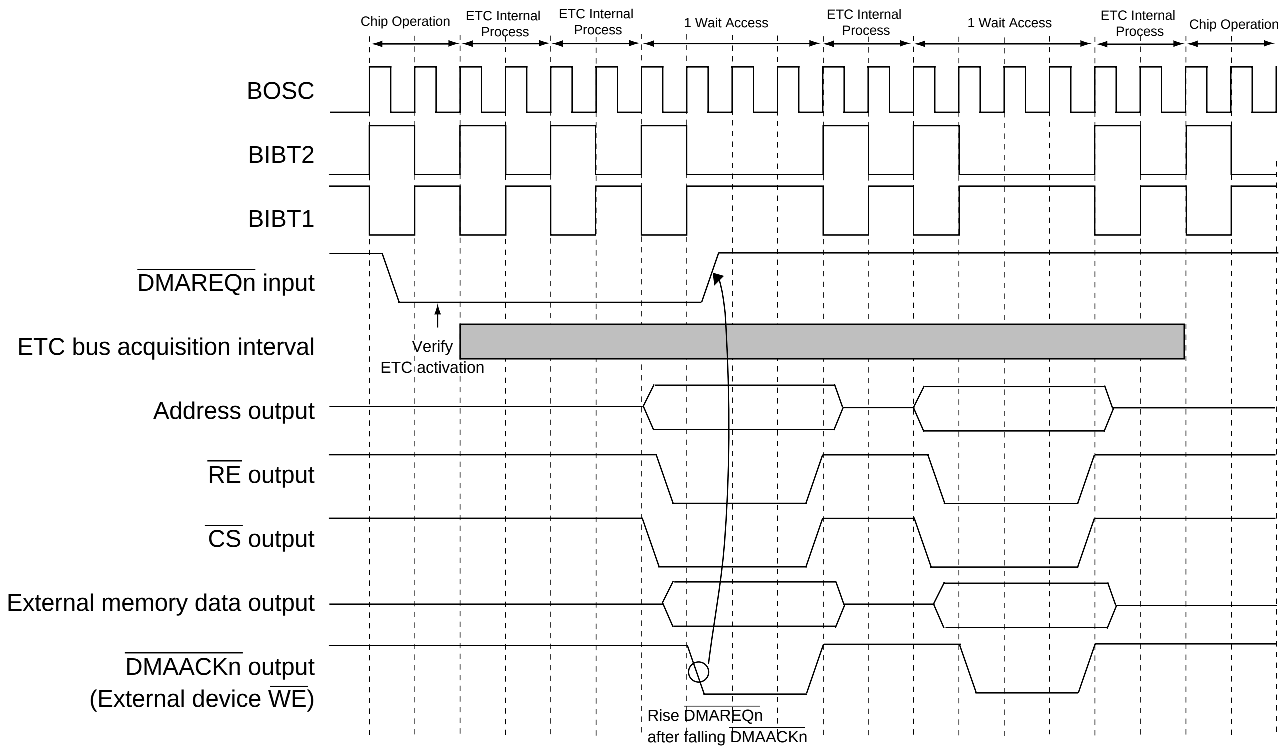
ET0ICL: x'00FDA5'

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	ET0 EN
0	0	0	0	0	0	0	1

The interrupt level is set in SC4TLV[2:0] of the SC4TICH register.

Under this state, ETC0 starts transferring when $\overline{\text{DMAREQ0}}$ becomes low by bus master. After the ETC0 transfer ends, an ETC0 transfer end interrupt occurs. Each ETC0 register value is set as follows:

ET0CNT: x'0FFF' (This value is always set regardless the bytes to be transferred.)
ET0SRC: x'100002' (The result incremented by 1 is set after the last transfer is completed.)



Note: the number of external memory waits = 1, the number of data transfer bytes =2

Figure 7-4-3 ETC External Memory → External Device Burst Transfer Timing

7-4-2 Transfer from External Device to External Memory (Burst Transfer)

/DMAREQ0 input from bus master is an activation factor. Each 16 bits of 4-byte data are transferred from the external device to the external memory.

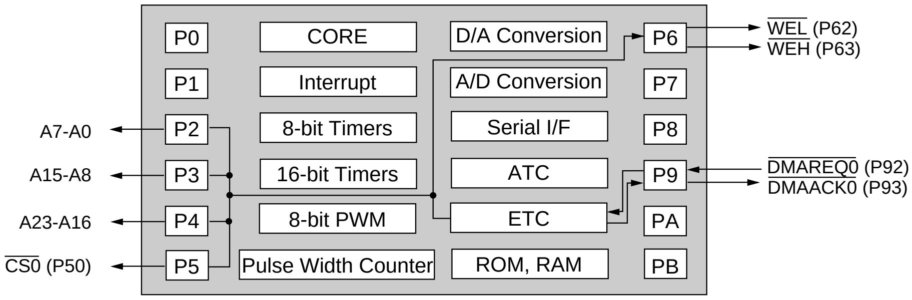


Figure 7-4-4 ETC External Device → External Memory (Burst)Transfer Block Diagram

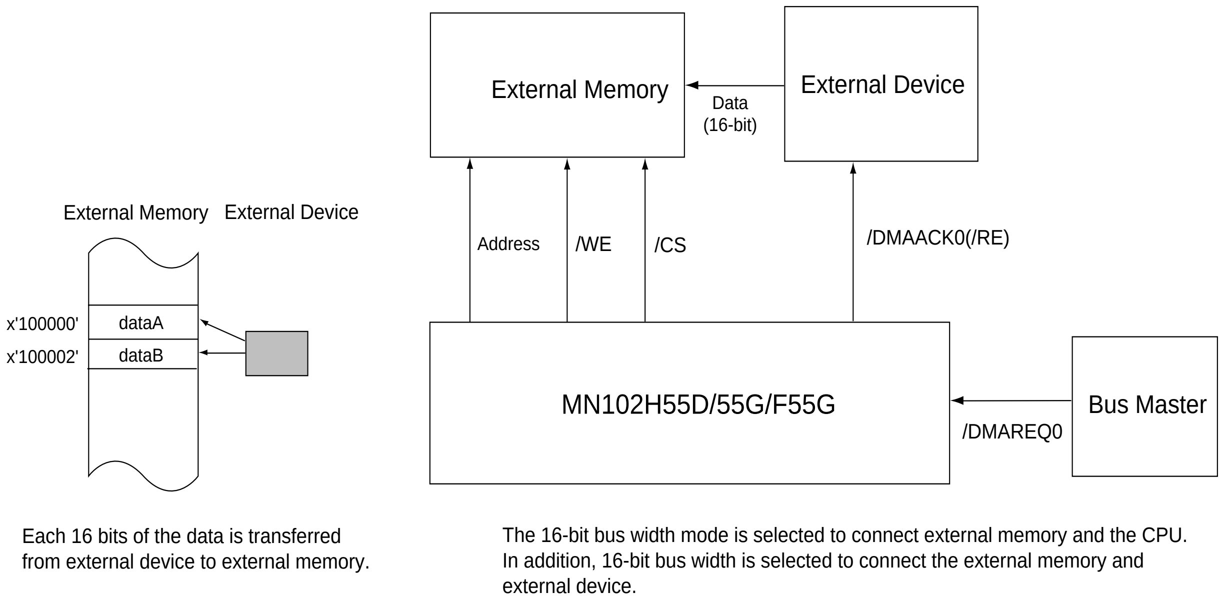


Figure 7-4-5 ETC External Device → External Memory (Burst) Transfer Connection

■ ETC Setup

(1) Set the destination address x'100000' of the external memory to the ETC0 destination address pointer (ET0SRC).

ET0DST: x'00FD48'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ET0 DST15	ET0 DST14	ET0 DST13	ET0 DST12	ET0 DST11	ET0 DST10	ET0 DST9	ET0 DST8	ET0 DST7	ET0 DST6	ET0 DST5	ET0 DST4	ET0 DST3	ET0 DST2	ET0 DST1	ET0 DST0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	ET0 DST23	ET0 DST22	ET0 DST21	ET0 DST20	ET0 DST19	ET0 DST18	ET0 DST17	ET0 DST16
								0	0	0	1	0	0	0	0

(2) Set the words to be transferred automatically. In this example, 2-word data is transferred so that the value '1' subtracting 2 by 1 is set to the ETC0 transfer word count register (ET0CNT).

ET0CNT: x'00FD42'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	ET0 CNT11	ET0 CNT10	ET0 CNT9	ET0 CNT8	ET0 CNT7	ET0 CNT6	ET0 CNT5	ET0 CNT4	ET0 CNT3	ET0 CNT2	ET0 CNT1	ET0 CNT0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

(3) Set the ETC0 control register (ET0CTR). Select burst transfer mode. Select one byte unit and the destination pointer to increment by 1. Select the transfer direction is from external device to external memory. Set the transfer start/busy flag to disable.

ET0CTR: x'00FD40'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ET0 FLG	-	ET0 MD0	ET0 BW	ET0 DB8	ET0 DI	ET0 SB8	ET0 SI	-	-	-	-	-	-	ET0 DIR	ET0 EN
0	0	1	0	0	1	0	0	0	0	0	0	0	0	0	1

■ Interrupt Setup

(4) Enable an ETC0 transfer end interrupt.

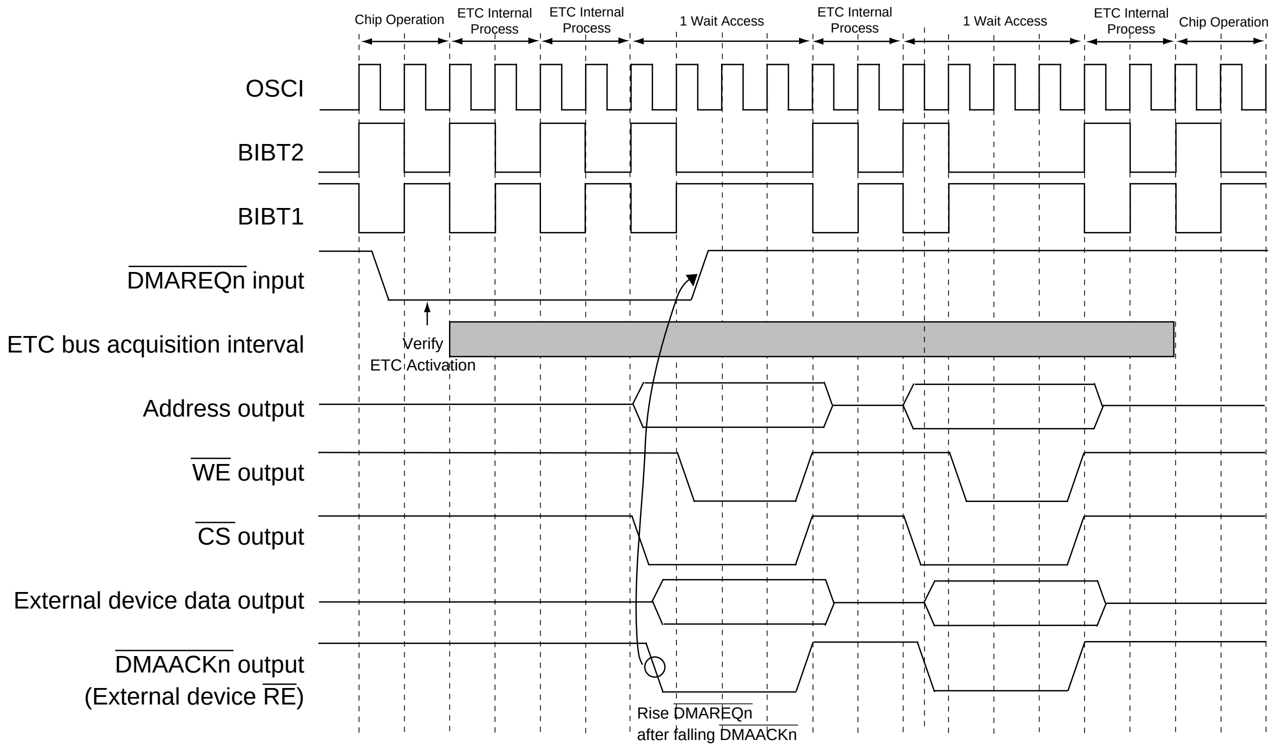
ET0ICL: x'00FDA5'

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	ET0 EN
0	0	0	0	0	0	0	1

The interrupt level is set in SC4TLV[2:0] of the SC4TICH register.

Under this state, ETC0 starts transferring when $\overline{\text{DMAREQ0}}$ becomes low by bus master. After the ETC0 transfer ends, an ETC0 transfer end interrupt occurs. Each ETC0 register value is set as follows:

ET0CNT: x'0FFF' (This value is always set regardless of the bytes to be transferred.)
ET0DST: x'100004' (The result incremented by 1 is set after the last transfer is completed.)



Note: the number of external memory waits =1, the number of data transfer bytes =2

Figure 7-4-6 ETC External Device → External Memory Burst Transfer Timing

7-4-3 Transfer from External Device to External Memory (One Byte Transfer)

/DMAREQ0 input from bus master is an activation factor. The two-byte data is transferred from the external device to the external memory.

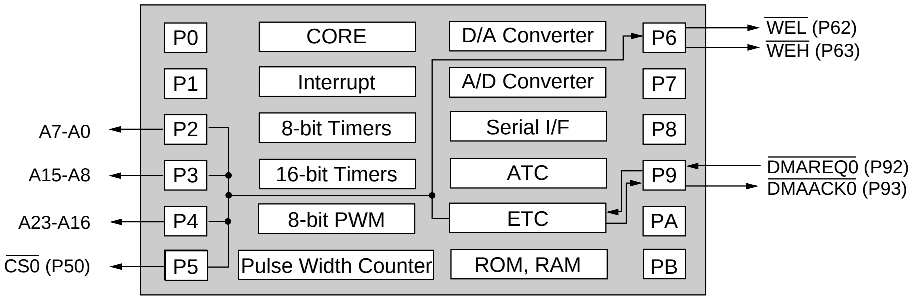


Figure 7-4-7 ETC External Device → External Memory (One Byte) Transfer Block Diagram

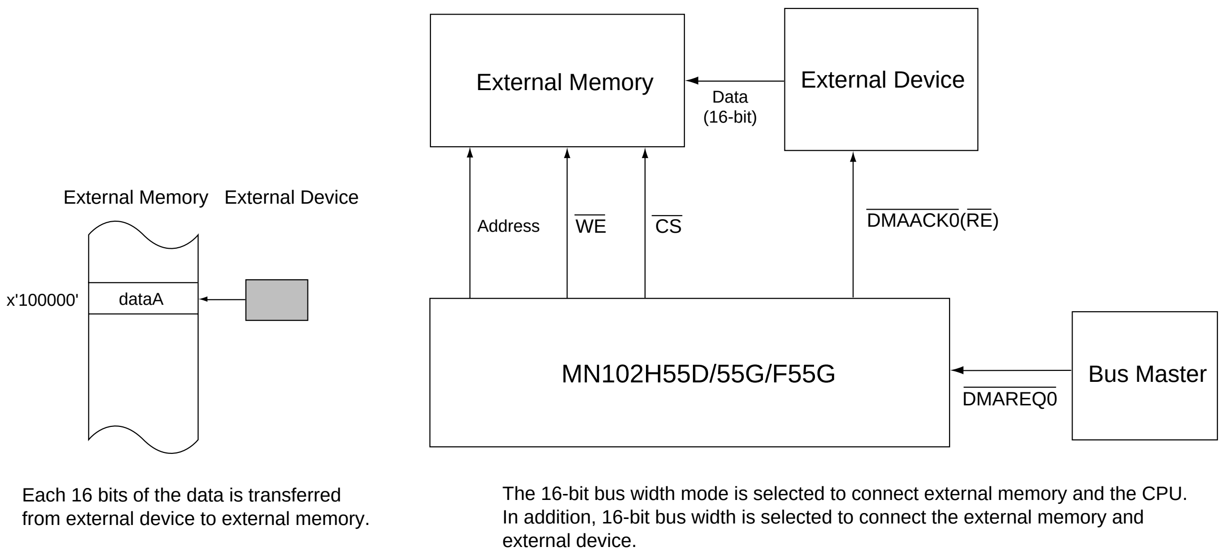


Figure 7-4-8 ETC External Device → External Memory (One Byte) Transfer Connection

■ ETC Setup

(1) Set the destination address x'100000' of the external memory to the ETC0 destination address pointer (ET0SRC).

ET0DST: x'00FD48'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ET0 DST15	ET0 DST14	ET0 DST13	ET0 DST12	ET0 DST11	ET0 DST10	ET0 DST9	ET0 DST8	ET0 DST7	ET0 DST6	ET0 DST5	ET0 DST4	ET0 DST3	ET0 DST2	ET0 DST1	ET0 DST0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	ET0 DST23	ET0 DST22	ET0 DST21	ET0 DST20	ET0 DST19	ET0 DST18	ET0 DST17	ET0 DST16
								0	0	0	1	0	0	0	0

(2) Set the words to be transferred automatically. In this example, 1-word data is transferred so that the value '0' subtracting 1 by 1 is set to the ETC0 transfer word count register (ET0CNT).

ET0CNT: x'00FD42'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	ET0 CNT11	ET0 CNT10	ET0 CNT9	ET0 CNT8	ET0 CNT7	ET0 CNT6	ET0 CNT5	ET0 CNT4	ET0 CNT3	ET0 CNT2	ET0 CNT1	ET0 CNT0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

(3) Set the ETC0 control register (ET0CTR). Select burst transfer mode. Select one word unit and the destination pointer to be fixed. Select the transfer direction is from external device to external memory. Set the transfer start/busy flag to enable.

ET0CTR: x'00FD40'

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ET0 FLG	-	ET0 MD0	ET0 BW	ET0 DB8	ET0 DI	ET0 SB8	ET0 SI	-	-	-	-	-	-	ET0 DIR	ET0 EN
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

■ Interrupt Setup

(4) Enable an ETC0 transfer end interrupt.

ET0ICL: x'00FDA5'

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	ET0 EN
0	0	0	0	0	0	0	1

The interrupt level is set in SC4TLV[2:0] of the SC4TICH register.

Under this state, ETC0 starts transferring when $\overline{\text{DMAREQ0}}$ becomes low by bus master. After the ETC0 transfer ends, an ETC0 transfer end interrupt occurs. Each ETC0 register value is set as follows:

ET0CNT: x'0FFF' (This value is always set regardless of the bytes to be transferred.)
ET0DST: x'100000' (The result incremented by 1 is set after the last transfer is completed.)

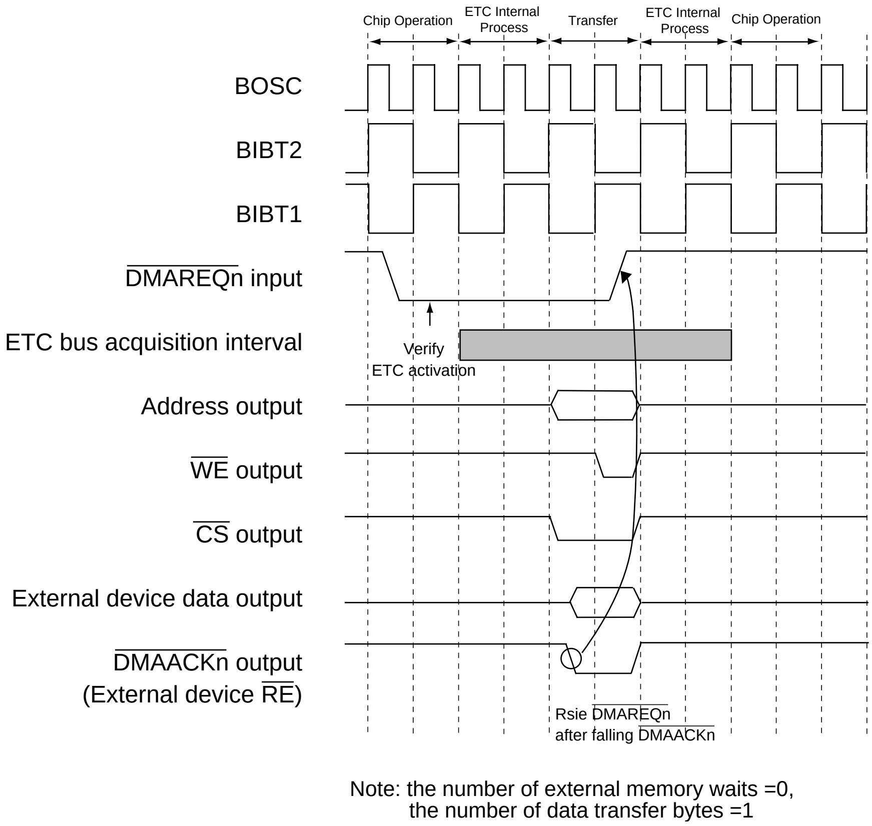


Figure 7-4-9 ETC External Device → External Memory (One Byte) Transfer Timing

Chapter 8 Ports

8

8-1 Summary of Ports

8-1-1 Overview

The MN102H55D/55G/F55G contains twelve I/O ports. Functions can be switched depending on the selected mode pins. Please refer to "11-2-3 Pin Functions" for details.

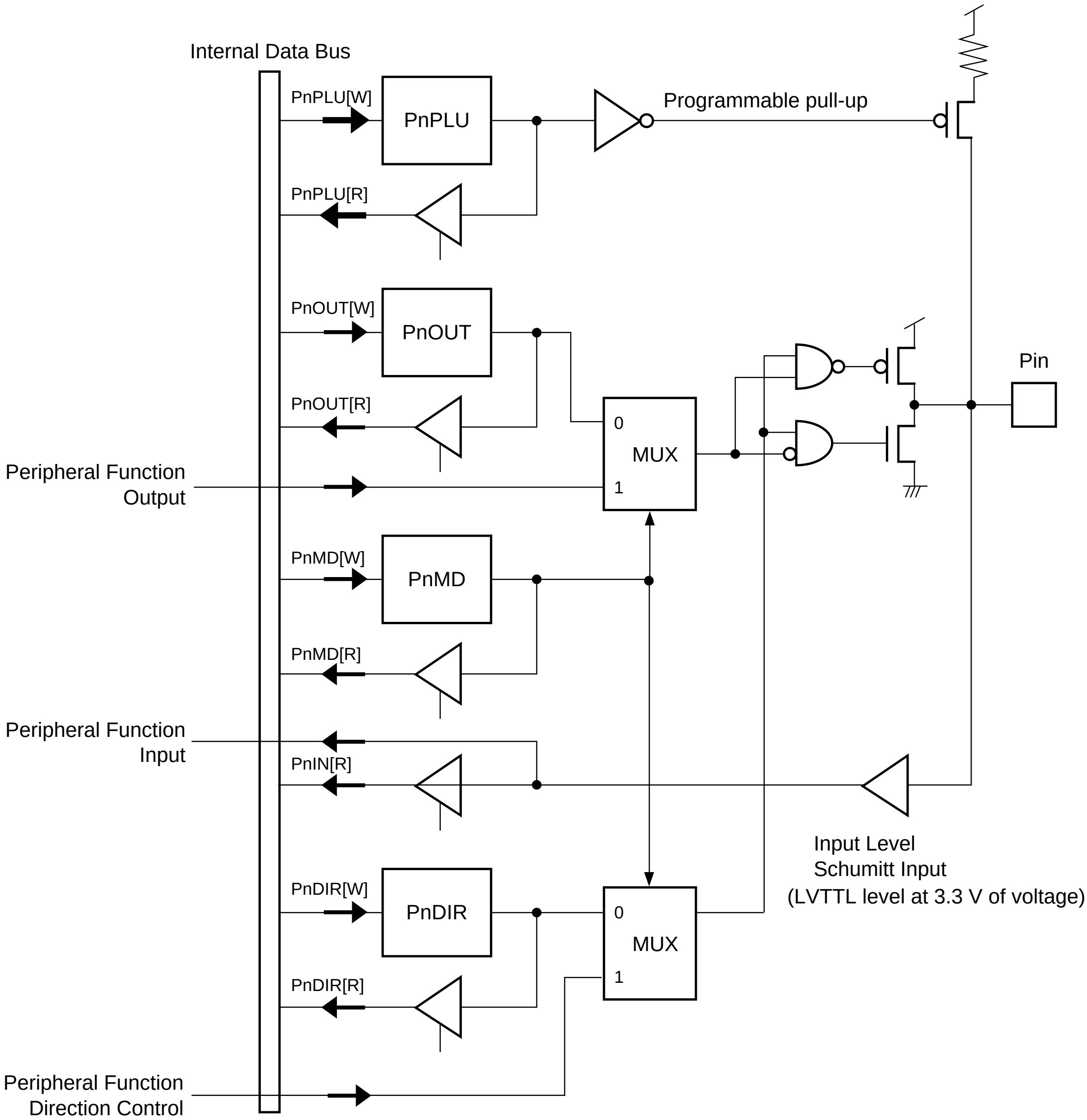


Figure 8-1-1 I/O Port Configuration

8-1-2 Control Registers

The MN102H55D/55G/F55G contains the port output register (PnOUT), the port input registers (PnIN), the port mode registers (PnMD), the port input/output control registers (PnDIR) and the port pull-up control registers (PnPLU). Refer to "11-2-3 List of Pin Functions" for details because some bits are not carried depending on ports. The port input/output control register is valid only when each port is used as its port input/output function. The direction is determined by setting each mode register when each port is used as an input/output pin of peripheral function.

PnOUT

	7	6	5	4	3	2	1	0
	Pn OUT7	Pn OUT6	Pn OUT5	Pn OUT4	Pn OUT3	Pn OUT2	Pn OUT1	Pn OUT0
Reset:	0	0	0	0	0	0	0	0

PnIN

	7	6	5	4	3	2	1	0
	Pn IN7	Pn IN6	Pn IN5	Pn IN4	Pn IN3	Pn IN2	Pn IN1	Pn IN0
Reset:	Port	Port	Port	Port	Port	Port	Port	Port

PnMD (PnLMD, PnMMD, PnHMD)

	7	6	5	4	3	2	1	0
	Pn MD7	Pn MD6	Pn MD5	Pn MD4	Pn MD3	Pn MD2	Pn MD1	Pn MD0
Reset:								

PnDIR

	7	6	5	4	3	2	1	0
	Pn DIR7	Pn DIR6	Pn DIR5	Pn DIR4	Pn DIR3	Pn DIR2	Pn DIR1	Pn DIR0
Reset:	0	0	0	0	0	0	0	0

0: Input
1: Output

PnPLU

	7	6	5	4	3	2	1	0
	Pn PLU7	Pn PLU6	Pn PLU5	Pn PLU4	Pn PLU3	Pn PLU2	Pn PLU1	Pn PLU0
Reset:	0	0	0	0	0	0	0	0

0: Pull-up off
1: Pull-up on

Table 8-1-1 List of Port Control Registers

Register	Address	R/W	Function
P0PLU	x'00FFB0'	R/W	Port 0 Pull-up Control Register
P0OUT	x'00FFC0'	R/W	Port 0 Output Register
P0IN	x'00FFD0'	R/W	Port 0 Input Register
P0DIR	x'00FFE0'	R/W	Port 0 Input/Output Control Register
P0MD	x'00FFF0'	R/W	Port 0 Mode Register
P1PLU	x'00FFB1'	R/W	Port 1 Pull-up Control Register
P1OUT	x'00FFC1'	R/W	Port 1 Output Register
P1IN	x'00FFD1'	R/W	Port 1 Input Register
P1DIR	x'00FFE1'	R/W	Port 1 Input/Output Control Register
P1LMD	x'00FFF2'	R/W	Port 1 Mode Register L
P1HMD	x'00FFF3'	R/W	Port 1 Mode Register H
P2PLU	x'00FFB2'	R/W	Port 2 Pull-up Control Register
P2OUT	x'00FFC2'	R/W	Port 2 Output Register
P2IN	x'00FFD2'	R/W	Port 2 Input Register
P2DIR	x'00FFE2'	R/W	Port 2 Input/Output Control Register
P2MD	x'00FFF1'	R/W	Port 2 Mode Register
P3PLU	x'00FFB3'	R/W	Port 3 Pull-up Control Register
P3OUT	x'00FFC3'	R/W	Port 3 Output Register
P3IN	x'00FFD3'	R/W	Port 3 Input Register
P3DIR	x'00FFE3'	R/W	Port 3 Input/Output Control Register
P3LMD	x'00FFF4'	R/W	Port 3 Mode Register L
P3HMD	x'00FFF5'	R/W	Port 3 Mode Register H
P4PLU	x'00FFB4'	R/W	Port 4 Pull-up Control Register
P4OUT	x'00FFC4'	R/W	Port 4 Output Register
P4IN	x'00FFD4'	R/W	Port 4 Input Register
P4DIR	x'00FFE4'	R/W	Port 4 Input/Output Control Register
P4LMD	x'00FFF6'	R/W	Port 4 Mode Register L
P4HMD	x'00FFF7'	R/W	Port 4 Mode Register H
P5PLU	x'00FFB5'	R/W	Port 5 Pull-up Control Register
P5OUT	x'00FFC5'	R/W	Port 5 Output Register
P5IN	x'00FFD5'	R/W	Port 5 Input Register
P5DIR	x'00FFE5'	R/W	Port 5 Input/Output Control Register
P5LMD	x'00FFF8'	R/W	Port 5 Mode Register L
P5HMD	x'00FFF9'	R/W	Port 5 Mode Register H
P6PLU	x'00FFB6'	R/W	Port 6 Pull-up Control Register
P6OUT	x'00FFC6'	R/W	Port 6 Output Register
P6IN	x'00FFD6'	R/W	Port 6 Input Register
P6DIR	x'00FFE6'	R/W	Port 6 Input/Output Control Register
P6MD	x'00FFFF'	R/W	Port 6 Mode Register

P7PLU	x'00FFB7'	R/W	Port 7 Pull-up Control Register
P7OUT	x'00FFC7'	R/W	Port 7 Output Register
P7IN	x'00FFD7'	R/W	Port 7 Input Register
P7DIR	x'00FFE7'	R/W	Port 7 Input/Output Control Register
P7LMD	x'00FFFA'	R/W	Port 7 Mode Register L
P7HMD	x'00FFFB'	R/W	Port 7 Mode Register H
P8PLU	x'00FFB8'	R/W	Port 8 Pull-up Control Register
P8OUT	x'00FFC8'	R/W	Port 8 Output Register
P8IN	x'00FFD8'	R/W	Port 8 Input Register
P8DIR	x'00FFE8'	R/W	Port 8 Input/Output Control Register
P8LMD	x'00FFFC'	R/W	Port 8 Mode Register L
P8MMD	x'00FFFD'	R/W	Port 8 Mode Register M
P8HMD	x'00FFFE'	R/W	Port 8 Mode Register H
P9PLU	x'00FFB9'	R/W	Port 9 Pull-up Control Register
P9OUT	x'00FFC9'	R/W	Port 9 Output Register
P9IN	x'00FFD9'	R/W	Port 9 Input Register
P9DIR	x'00FFE9'	R/W	Port 9 Input/Output Control Register
P9LMD	x'00FFEC'	R/W	Port 9 Mode Register L
P9HMD	x'00FFED'	R/W	Port 9 Mode Register H
PAPLU	x'00FFBA'	R/W	Port A Pull-up Control Register
PAOUT	x'00FFCA'	R/W	Port A Output Register
PAIN	x'00FFDA'	R/W	Port A Input Register
PADIR	x'00FFEA'	R/W	Port A Input/Output Control Register
PAMD	x'00FFDC'	R/W	Port A Mode Register
PBPLU	x'00FFBB'	R/W	Port B Pull-up Control Register
PBOUT	x'00FFCB'	R/W	Port B Output Register
PBIN	x'00FFDB'	R/W	Port B Input Register
PBDIR	x'00FFEB'	R/W	Port B Input/Output Control Register
PBMD	x'00FFDD'	R/W	Port B Mode Register

8-1-3 Port Block Diagram

The MN102H55D/55G/F55G contains twelve I/O ports of P0 to PB. A set consists of two ports to eight ports. Each pin serves as a general-purpose port function or an input/output function for each peripheral function. The function can be switched by each port mode register. When the input/output pin of the peripheral function is selected, setting each port mode register determines the input/output direction automatically. When the general-purpose input/output port is selected, each port direction control register controls the input/output direction. Each port has a pull-up resistor which is controlled by software and switches ON/OFF regardless of each port mode register and the direction control register setup.

Table 8-1-2 Port Block Diagram (1/12)

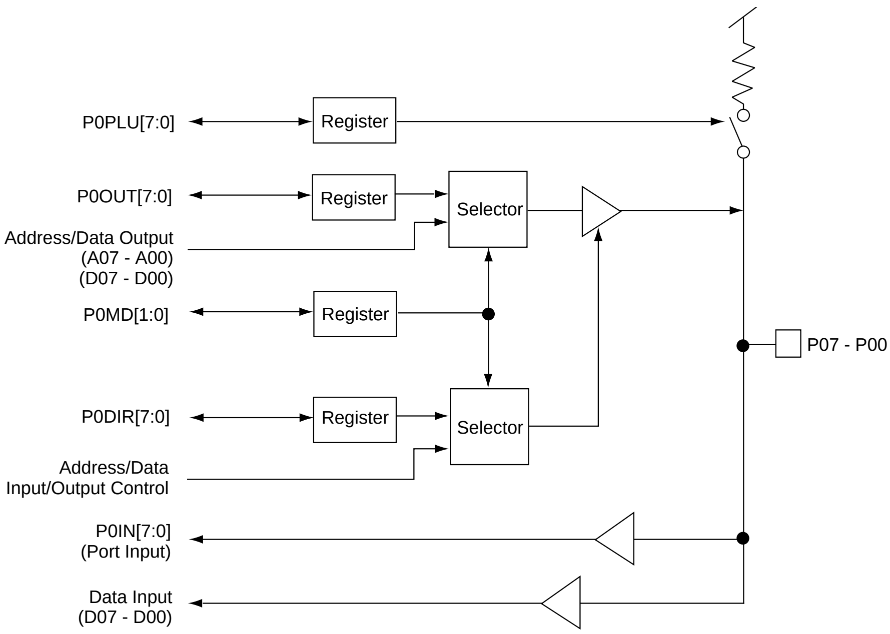
Port	Pin Name	Block Diagram
Port 0	P07 to P00 D07 to D00 A07 to A00	 Note: The set value of the P0DIR register is valid only when the port function is selected by the P0MD register.

Table 8-1-2 Port Block Diagram (2/12)

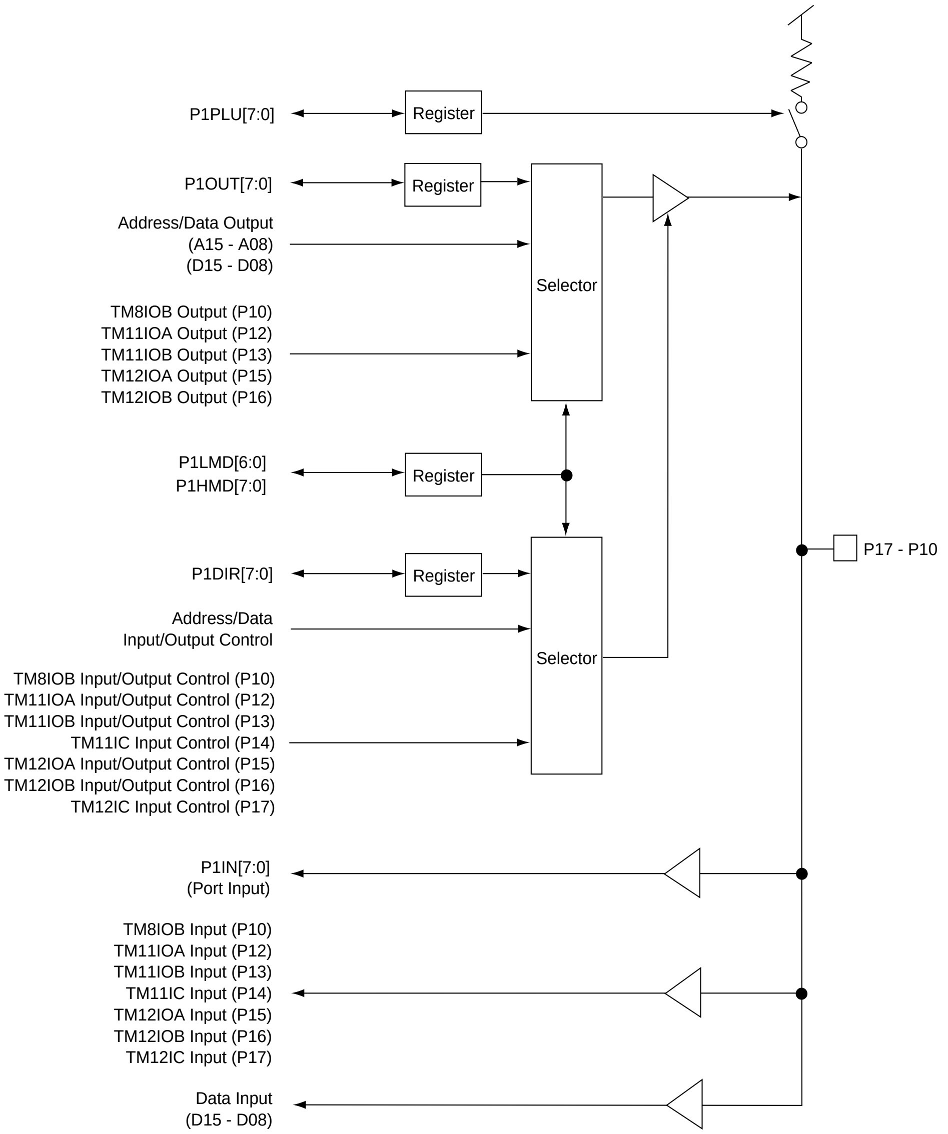
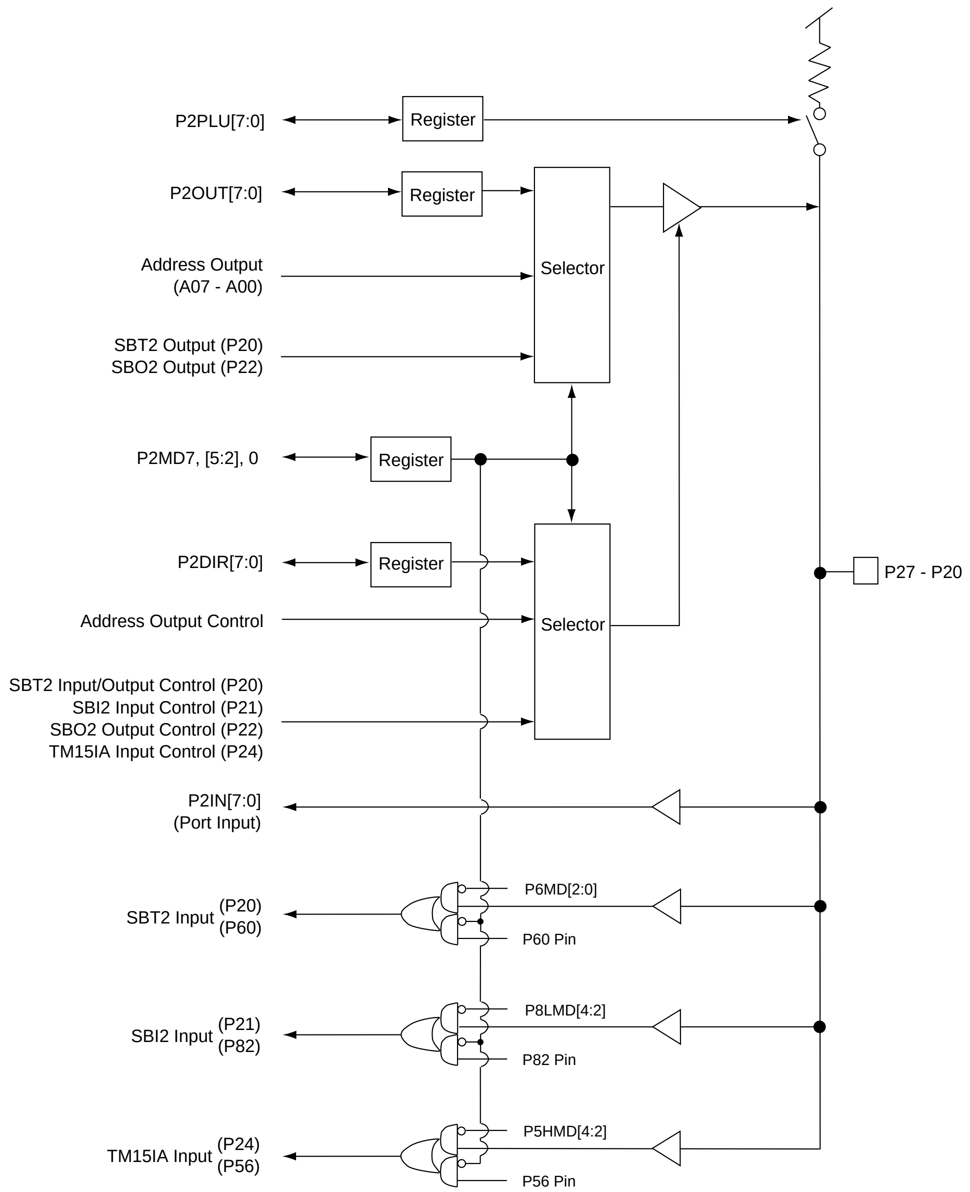
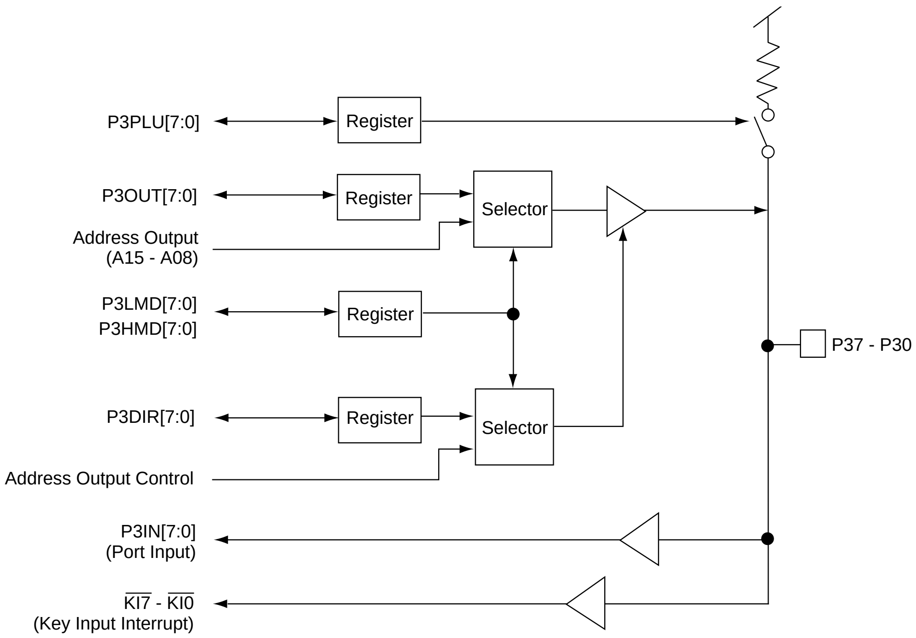
Port	Pin Name	Block Diagram
Port 1	P17 to P10 D15 to D08 A15 to A08 TM8IOB TM11IOA TM11IOB TM11IC TM12IOA TM12IOB TM12IC	 Note : The set value of the P1DIR register is valid only when the port function is selected by the P1LMD register or P1HMD register. The input or output direction of TMnIOA, TMnIOB and TMnIC is determined by setting the P1LMD register or P1HMD register.

Table 8-1-2 Port Block Diagram (3/12)

<table>
<tr><th>Port</th><th>Pin Name</th><th>Block Diagram</th></tr>
<tr><td>Port 2</td><td>P27 to P20
A07 to A00
SBT2
SBO2
SBI2
TM15IA</td><td>

Note : The set value of the P2DIR register is valid only when the port function is selected by the P2MD register.
The input or output direction of SBT2, SBI2, SBO2 and TM15IA is determined by setting the P2MD register.</td></tr>
</table>

Table 8-1-2 Port Block Diagram (4/12)

<table>
<tr><th>Port</th><th>Pin Name</th><th>Block Diagram</th></tr>
<tr><td>Port 3</td><td>P37 to P30
A15 to A08
$\overline{\text{KI7}}$ to $\overline{\text{KI0}}$</td><td>


Note : The set value of the P3DIR register is valid only when the port function is selected by the P3LMD register or the P3HMD register.
The input or output direction of $\overline{\text{KI7}}$ to $\overline{\text{KI0}}$ is determined automatically by setting the P3LMD register or the P3HMD register.
</td></tr>
</table>

Table 8-1-2 Port Block Diagram (5/12)

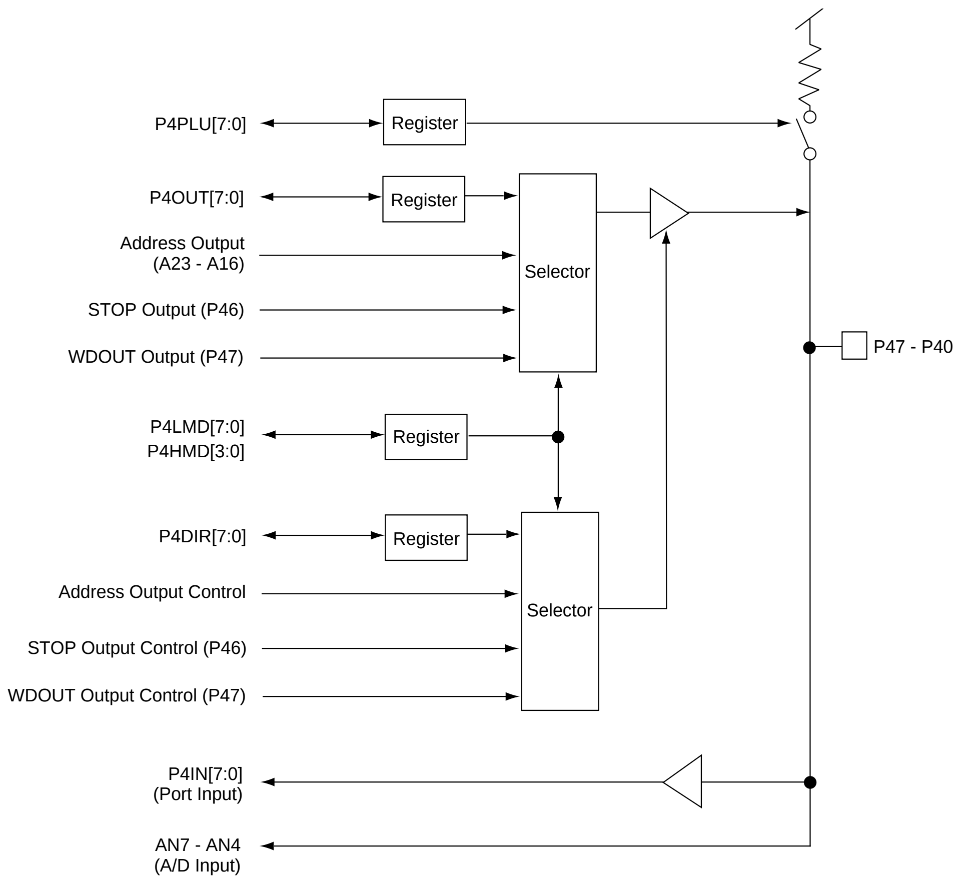
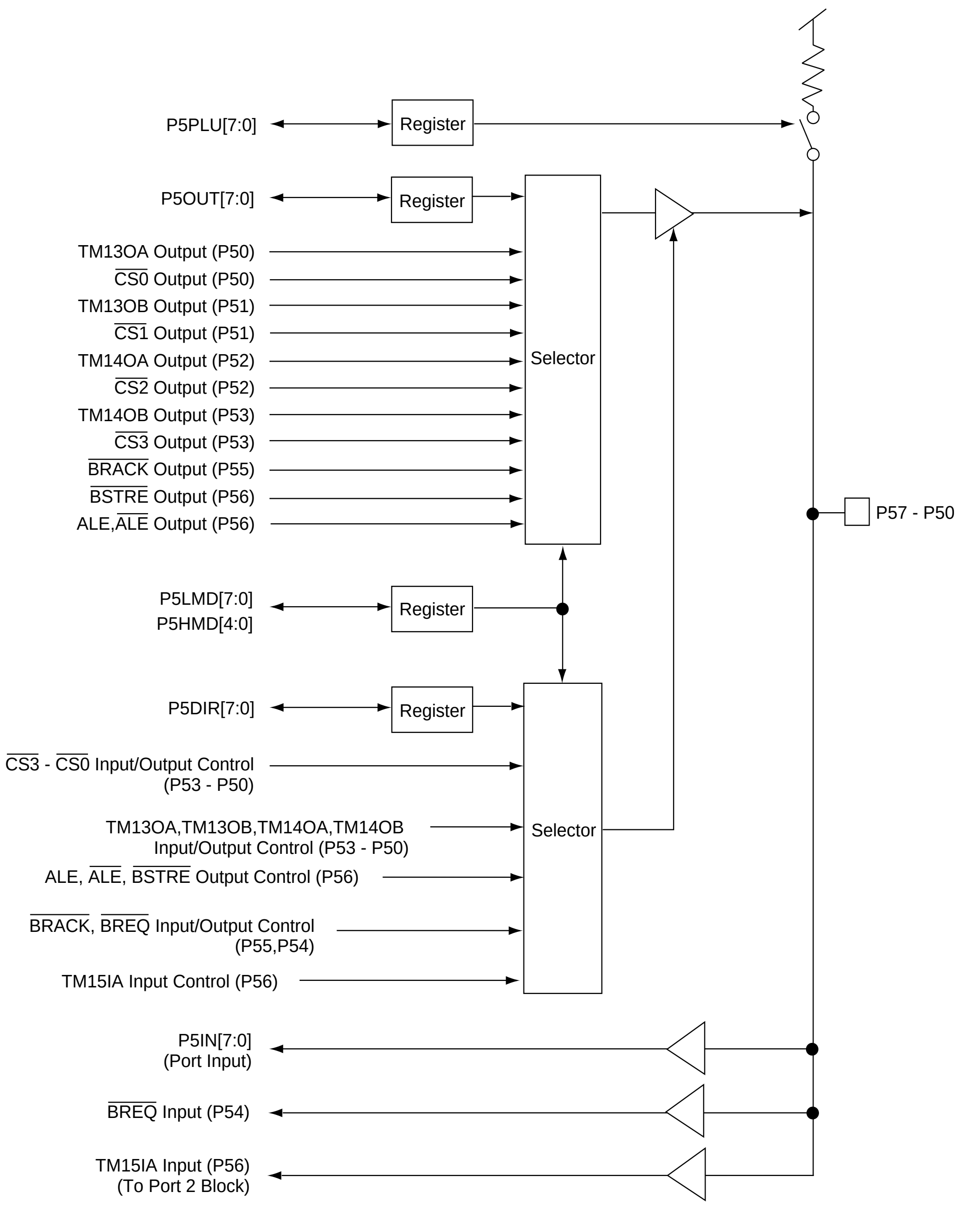
Port	Pin Name	Block Diagram
Port 4	P47 to P40 A23 to A16 STOP WDOUT AN7 to AN4	 Note : The set value of the P4DIR register is valid only when the port function is selected by the P4LMD register or the P4HMD register. The input or output direction of STOP and WDOUT is determined automatically by setting the P4LMD register or the P4HMD register.

Table 8-1-2 Port Block Diagram (6/12)

<table>
<tr><th>Port</th><th>Pin Name</th><th>Block Diagram</th></tr>
<tr><td>Port 5</td><td>P57 to P50
TM13OA
TM13OB
TM14OA
TM14OB
TM15IA
$\overline{\text{CS3}}$ to $\overline{\text{CS0}}$
$\overline{\text{BREQ}}$
$\overline{\text{BRACK}}$
$\overline{\text{BSTRE}}$
ALE
$\overline{\text{ALE}}$</td><td>



Note : The set value of the P5DIR register is valid only when the port function is selected by the P5LMD register or the P5HMD register.
The input or output direction of $\overline{\text{CS3}}$ to $\overline{\text{CS0}}$, ALE, $\overline{\text{ALE}}$, $\overline{\text{BREQ}}$, $\overline{\text{BRACK}}$ and timer output is determined automatically by setting the P5LMD register or the P5HMD register.
</td></tr>
</table>

Table 8-1-2 Port Block Diagram (7/12)

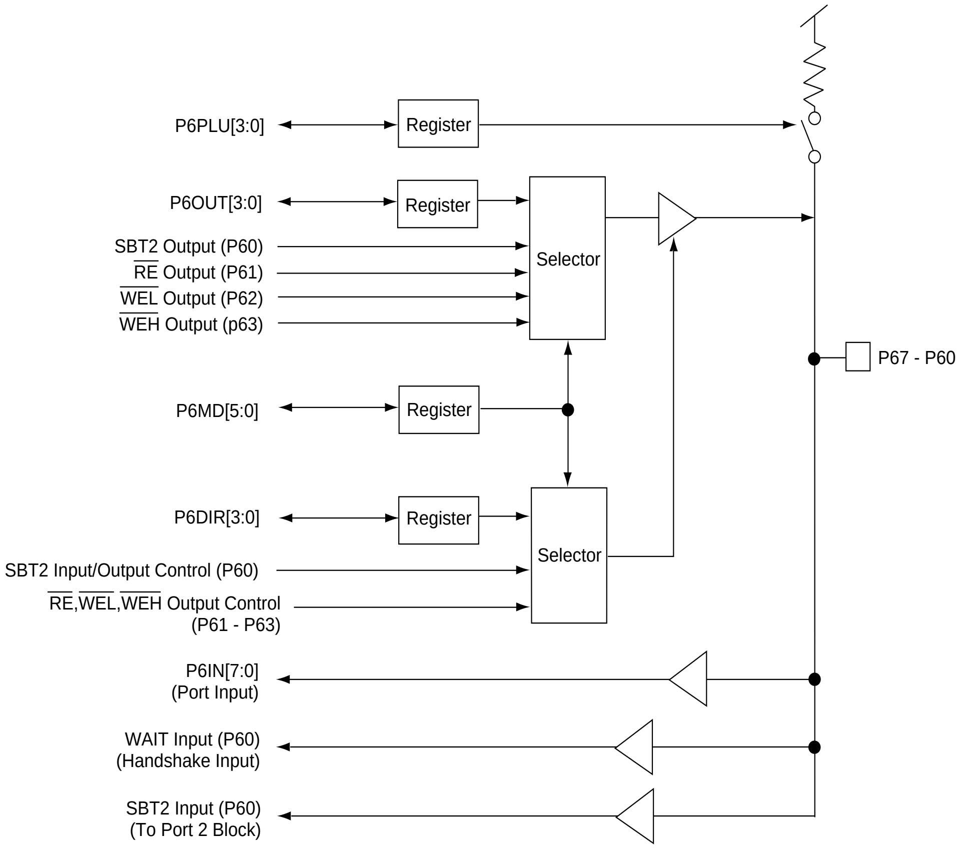
Port	Pin Name	Block Diagram
Port 6	P63 to P60 SBT2 $\overline{\text{RE}}$ $\overline{\text{WEL}}$ $\overline{\text{WEH}}$ WAIT	 Note : The set value of the P6DIR register is valid only when the port function is selected by the P6MD register. The input or output direction of SBT2, $\overline{\text{RE}}$, $\overline{\text{WEL}}$, $\overline{\text{WEH}}$ and WAIT is determined automatically by setting the P6MD register.

Table 8-1-2 Port Block Diagram (8/12)

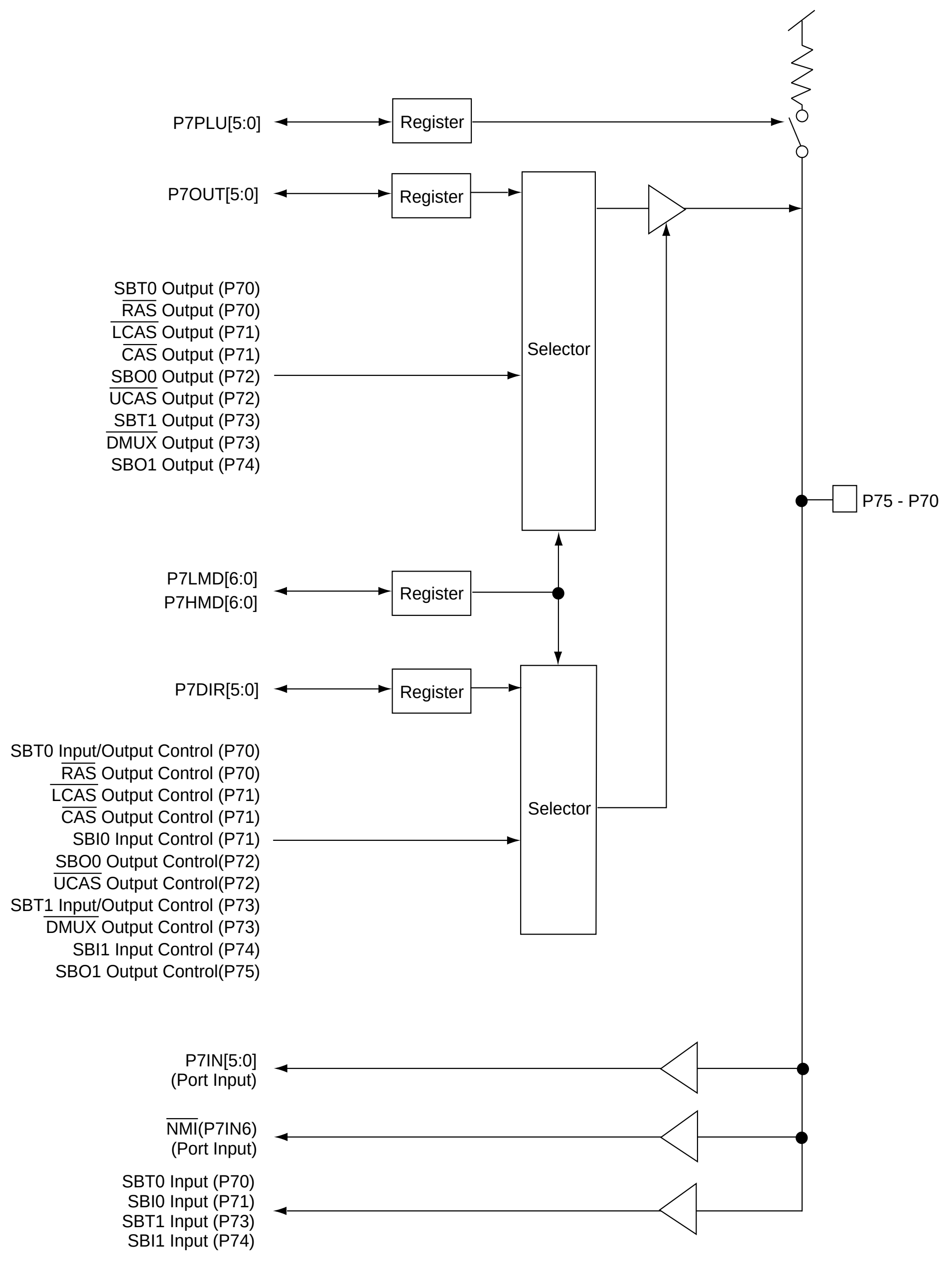
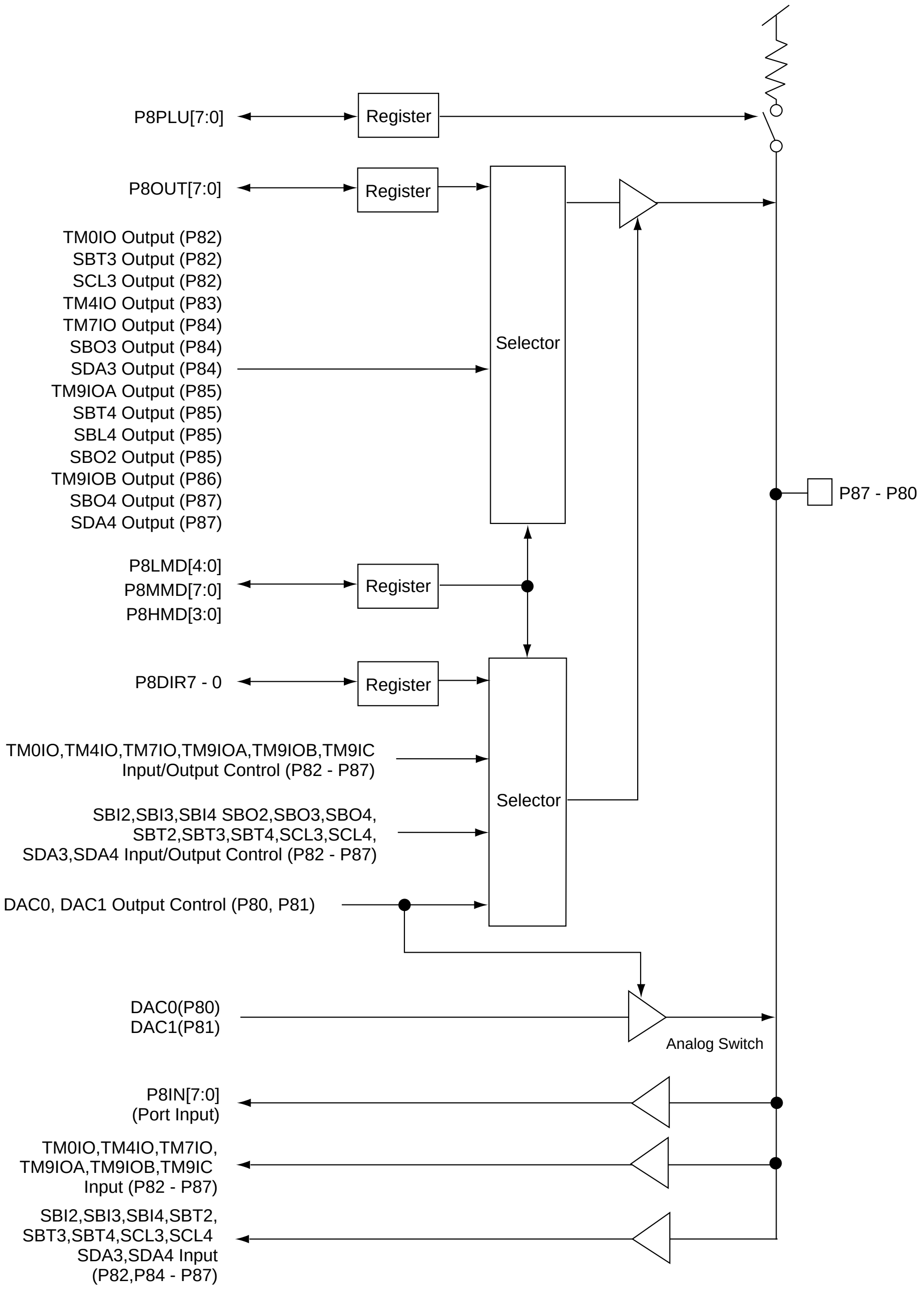
Port	Pin Name	Block Diagram
Port 7	P75 to P70 SBT0 SBI0 SBO0 SBT1 SBI1 SBO1 $\overline{\text{RAS}}$ $\overline{\text{LCAS}}$ $\overline{\text{CAS}}$ $\overline{\text{UCAS}}$ $\overline{\text{DMUX}}$ $\overline{\text{NMI}}$ (P76)	 Note : The set value of the P7DIR register is valid only when the port function is selected by the P7LMD register or the P7HMD register. The input or output direction of serial and DRAM I/F is determined automatically by setting the P7LMD register or P7HMD register.

Table 8-1-2 Port Block Diagram (9/12)

<table>
<tr><th>Port</th><th>Pin Name</th><th>Block Diagram</th></tr>
<tr><td>Port 8</td><td>P87 to P80
DAC0
DAC1
TM0IO
TM4IO
TM7IO
TM9IOA
TM9IOB
TM9IC
SBI2
SBO2
SBT3
SCL3
SBI3
SBO3
SDA3
SBT4
SCL4
SBI4
SBO4
SDA4</td><td>


Note : The set value of the P8DIR register is valid only when the port function is selected by the P8LMD register, the P8HMD register.
The input or output direction of D/A function, timer function and serial function is determined automatically by setting the P8LMD register or the P8HMD register.
</td></tr>
</table>

Table 8-1-2 Port Block Diagram (10/12)

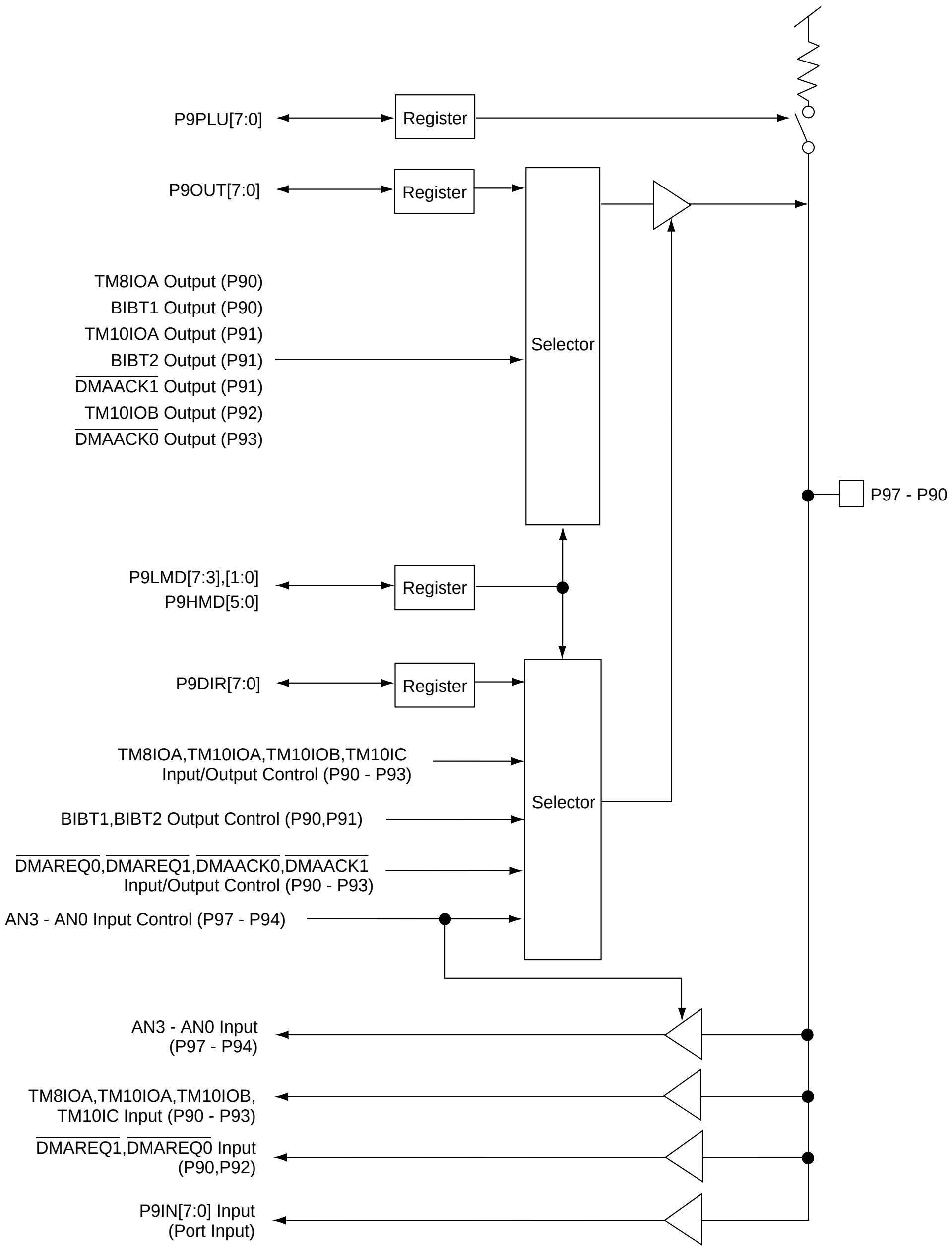
Port	Pin Name	Block Diagram
Port 9	P97 to P90 TM8IOA TM10IOA TM10IOB TM10IC BIBT1 BIBT2 $\overline{\text{DMAREQ1}}$ $\overline{\text{DMAREQ0}}$ $\overline{\text{DMAACK1}}$ $\overline{\text{DMAACK0}}$ AN3 to AN0	 Note : The set value of the P9DIR register is valid only when the port function is selected by the P9LMD register or the P9HMD register. The input or output direction of timer, BIBT1, BIBT2, ETC and A/D is determined automatically by setting the P9LMD register or P9HMD register.

Table 8-1-2 Port Block Diagram (11/12)

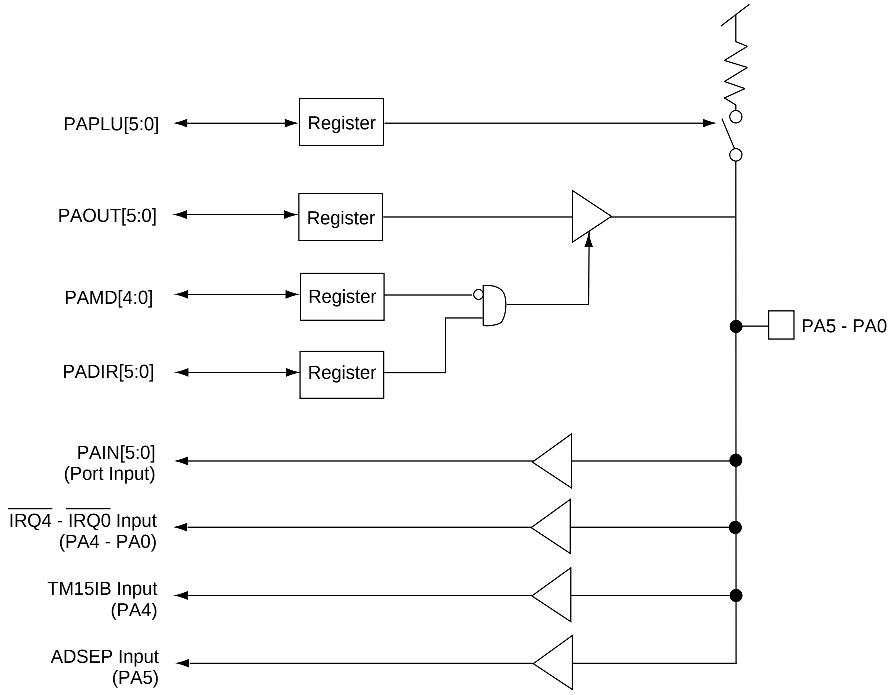
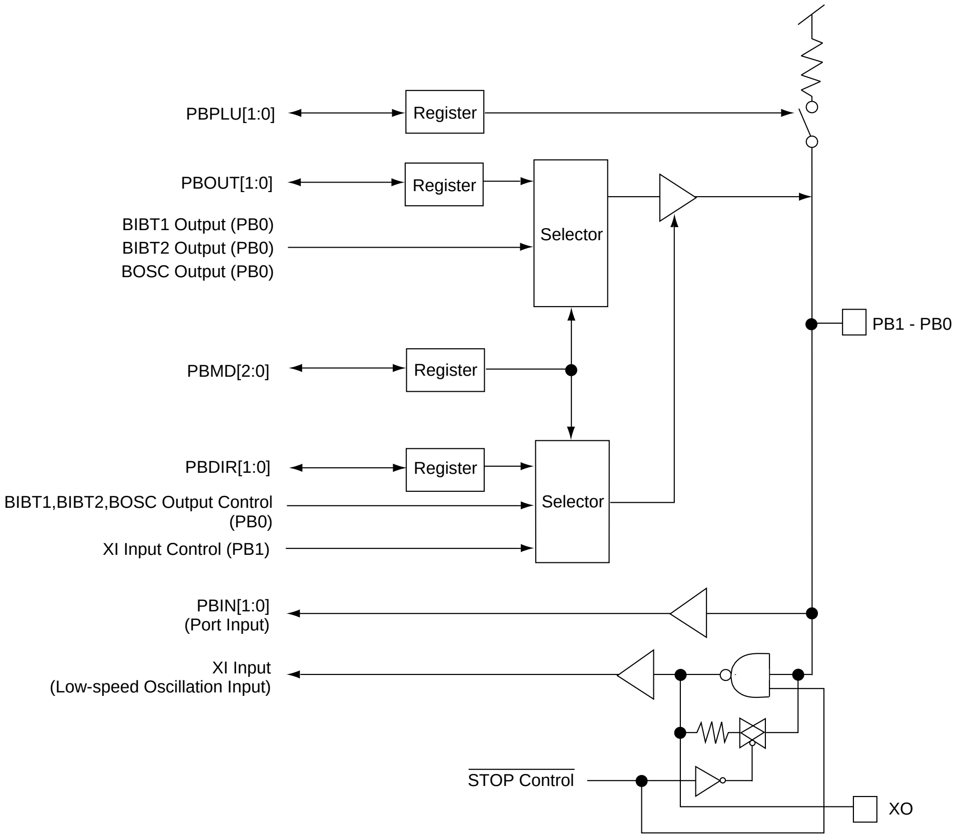
Port	Pin Name	Block Diagram
Port A	PA5 to PA0 $\overline{IRQ4}$ to $\overline{IRQ0}$ TM15IB ADSEP	 Note : The set value of the PADIR register is valid only when the port function is selected by the PAMD register. The input or output direction of interrupt and timer function is determined automatically by setting the PAMD register.

Table 8-1-2 Port Block Diagram (12/12)

Port	Pin Name	Block Diagram
Port B	PB1 to PB0 BIBT1 BIBT2 BOSC XI XO	 Note : The set value of the PBDIR register is valid only when the port function is selected by the PBMD register. The input or output direction of BIBT1, BIBT2, BOSC and XI is determined automatically by setting the PBMD register. The XI pin can be used as the port B1 when this pin is not used as the low-speed oscillation pin.

8-2 Port Setup Examples

8-2-1 General-purpose Port Setup

This section describes a light-emitting diode (LED) on/off based on switch input status. P71 is connected to the switch and P70 is connected to the LED. In this configuration, the LED is on when the switch is on while the LED is off when the switch is off.

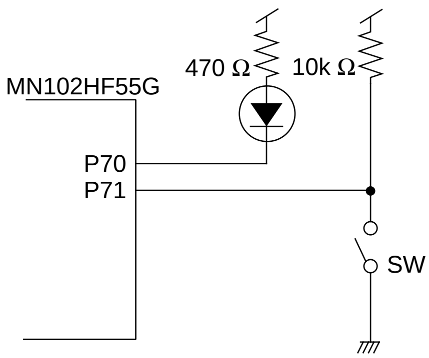


Figure 8-2-1 General-purpose Port Setup Example

(1) Both P71 pin and P70 pin are set to input by the initial values after reset release. Under this condition, the LED is off. Next, set the P70 pin to the general-purpose port output.

P7DIR: x'00FFE7'

7	6	5	4	3	2	1	0
P7 DIR7	P7 DIR6	P7 DIR5	P7 DIR4	P7 DIR3	P7 DIR2	P7 DIR1	P7 DIR0
0	0	0	0	0	0	0	1

P7LMD: x'00FFFA'

7	6	5	4	3	2	1	0
P7 LMD7	P7 LMD6	P7 LMD5	P7 LMD4	P7 LMD3	P7 LMD2	P7 LMD1	P7 LMD0
0	0	0	0	0	0	0	0

(2) Read the P71 pin status (P7IN) with the MOVB instruction. If bit 1 is '0', set P0OUT to x'00'.

P7OUT: x'00FFC7'

7	6	5	4	3	2	1	0
P7 OUT7	P7 OUT6	P7 OUT5	P7 OUT4	P7 OUT3	P7 OUT2	P7 OUT1	P7 OUT0
0	0	0	0	0	0	0	0

P7IN: x'00FFD7'

7	6	5	4	3	2	1	0
P7 IN7	P7 IN6	P7 IN5	P7 IN4	P7 IN3	P7 IN2	P7 IN1	P7 IN0

On the contrary, if bit 1 is '1', set P7OUT to x'01'.

P7OUT: x'00FFC7'

7	6	5	4	3	2	1	0
P7 OUT7	P7 OUT6	P7 OUT5	P7 OUT4	P7 OUT3	P7 OUT2	P7 OUT1	P7 OUT0
0	0	0	0	0	0	0	1

Under this condition, the low level is output to P70 pin if the switch is on while the high level is output to P70 pin if the switch is off resulting that the light-emitting diode is on or off. Thereafter, reading the P71 pin status is repeated.

Figure 8-2-2 and Figure 8-2-3 show the flowcharts of general-purpose port operations. When the port is input, set the PnMDm flag and PnDIRm flag to '0' and read the PnINm flag. When the port is output, set PnDIRm flag to '1' and write the data output to the PnOUTm flag. Regardless of input or output direction, set the PnPLUm flag to '1' for the pull-up setting. (n means the port number, m means bit position.)

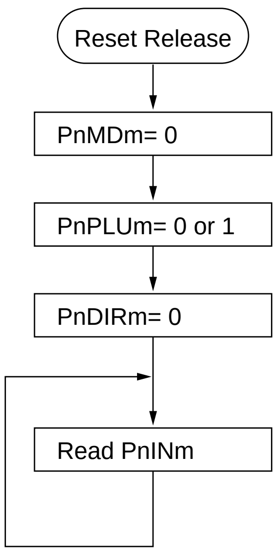


Figure 8-2-2 Basic Flowchart of General-purpose Port Input

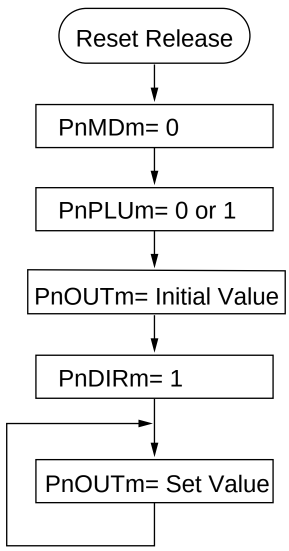


Figure 8-2-3 Basic Flowchart of General-purpose Port Output

8-3 Summary of Byte-swapped Registers

8-3-1 Overview

The MN102H55D/55G/F55G contains byte-swapped registers for pointers and long-word data. Each written data is swapped and read as follows.

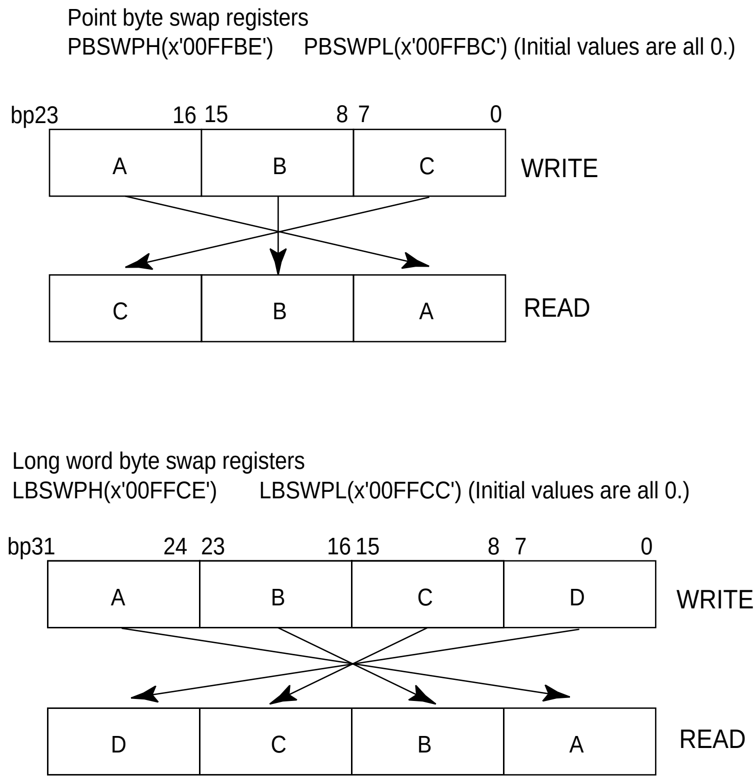


Figure 8-3-1 Byte-swapped Register

The MN102H55D/55G/F55G has no byte-swapped registers for the word data. When the word data needs to be swapped, use the byte-swapped register for long-word data. Write the word data to LBSWPH (x'00FFCE') and read the data from LBSWPL (x'00FFCC') or vice versa.

Chapter 9 System Control

9

9-1 Address Break

9-1-1 Overview

In the MN102HF55G (Flash ROM version) or ICE, this function cannot be used. In addition, in the MN102H55D/55G, the address break 0/1 generation flags of the address break control register cannot be used. Instead, the function of these flags are substituted by verifying whether the PC values (upper or lower bits) on the stack pointer match the address break 0/1 address pointer during $\overline{\text{NMI}}$ interrupt routine service.

The MN102H55D/55G/F55G generates a $\overline{\text{NMI}}$ interrupt before executing the instruction located on an arbitrary address. The MN102H55D/55G/F55G has two registers of the address break 0 address pointer and the address break 1 address pointer specifying the address where an interrupt is generated. When the address of the instruction fetch matches the address of either the address break 0 address pointer or the address break 1 address pointer, the CPU generates a $\overline{\text{NMI}}$ interrupt by replacing its instruction code into the undefined instruction (x'FF'). This function can debug the software or correct mask ROM under the production process.

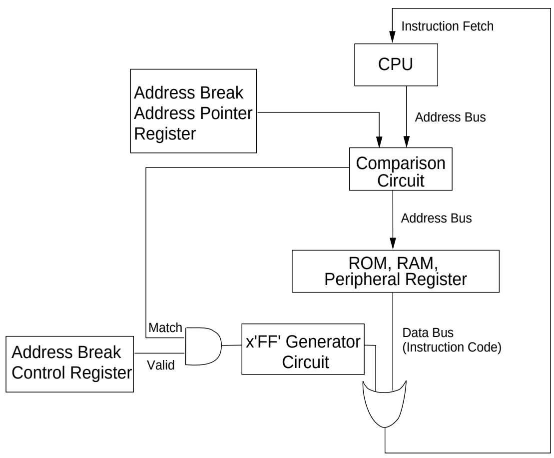


Figure 9-1-1 Address Break Operation Example

Set the first address of the instruction code to be suspended to the address break address pointer.

The address break function makes the CPU suspend executing all instructions.

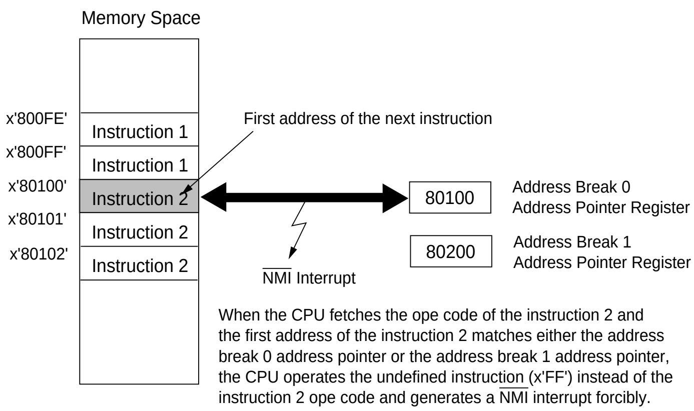


Figure 9-1-2 Address Break Block Diagram

9-1-2 Control Registers

The MN102H55D/55G/F55G contains the address break address pointers (ADBn) and the address break control register (ADBCTL).

ADBCTL: x'00FCDA'

Address Break Control Register

	7	6	5	4	3	2	1	0
	—	—	—	—	ADB1 ON	ADB0 ON	ADB1 CK	ADB0 CK
At reset:	0	0	0	0	0	0	0	0

ADB0CK	Address Break 0 Generation
0	Do not generate
1	Generated

ADB1CK	Address Break 1 Generation
0	Do not generate
1	Generated

ADB0ON	Address Break 0 Operation
0	OFF
1	ON

ADB1ON	Address Break 1 Operation
0	OFF
1	ON

In the MN102H55D/55G, the ADB0CK and ADB1CK flags do not operate correctly. Compare the return address to the ADB0 register or ADB1 register to check the address break interrupt generation. The return address is the ADB0 register value plus 1 or the ADB1 register value plus 1.

ADB0: x'00FCD2'
ADB1: x'00FCD6'

Address Break n Address Pointer

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADBn A15	ADBn A14	ADBn A13	ADBn A12	ADBn A11	ADBn A10	ADBn A9	ADBn A8	ADBn A7	ADBn A6	ADBn A5	ADBn A4	ADBn A3	ADBn A2	ADBn A1	ADBn A0
At reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	-	-	ADBn A23	ADBn A22	ADBn A21	ADBn A20	ADBn A19	ADBn A18	ADBn A17	ADBn A16
At reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

9-1-3 Address Break Setup Examples

When an error occurs in the routine on the internal ROM, the program cannot be corrected normally. An error, however, can be avoided by storing the solution program on the internal RAM or the external RAM and setting the address break.

For example, the CPU stores the address where the instruction execution is halted and the substitute program on the nonvolatile memory connected externally to the MN102H55D/55G/F55G. Then the CPU accesses the nonvolatile memory from the main routine after reset. Finally, the CPU loads the address where the instruction execution is halted and the substitute program on the register or the internal RAM when the address break function is required. In this example, execute the subroutine 2 on the internal RAM without executing the subroutine 1 on the internal ROM.

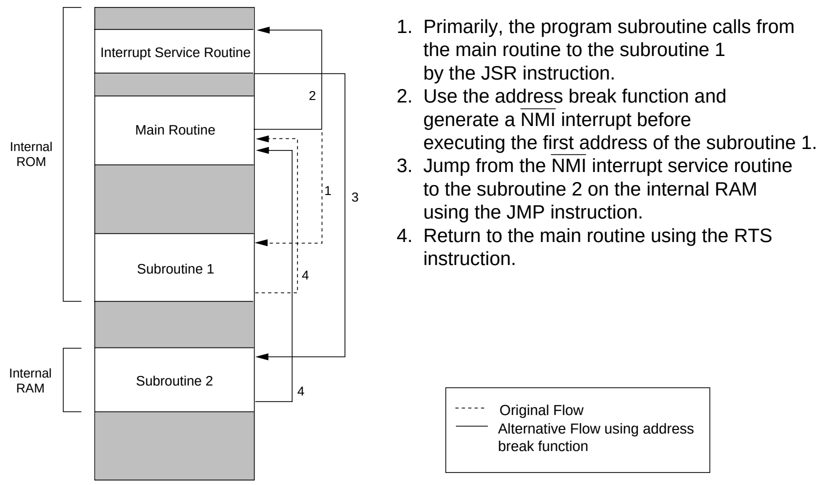


Figure 9-1-3 Program Flow of Address Break Setup

■ **Address Break Setup**

(1) Set the first address of the subroutine to the address break 0 address pointer (ADB0).

(2) Set 1 to the ADB0ON bit of the address break control register.

ADBCTL: x'00FCDA'

7	6	5	4	3	2	1	0
-	-	-	-	ADB1 ON	ADB0 ON	ADB1 CK	ADB0 CK
0	0	0	0	0	1	0	0

Thereafter, a $\overline{\text{NMI}}$ interrupt occurs when the CPU executes the address set in the step (1).

■ $\overline{\text{NMI}}$ Interrupt Service Routine

In the MN102H55D/55G, the ADB0CK and ADB1CK flags do not operate correctly. Compare the return address to the ADB0 register or ADB1 register to check the address break interrupt generation. The return address is the ADB0 register value plus 1 or the ADB1 register value plus 1.

(3) Jump to the address x'80008' when the address break occurs. The value of the IAGR register at this point is 8. Verify that the ADB0CK flag of the address break control register (ADBCTL) is 1 during the interrupt service routine. This determines whether a $\overline{\text{NMI}}$ interrupt occurs by the address break or other factors. Clear the ADB0CK flag and the ADB1CK flag to '0' by software after verification because both the ADB0CK flag and the ADB1CK flag are not cleared automatically.

ADBCTL: x'00FCDA'

7	6	5	4	3	2	1	0
-	-	-	-	ADB1 ON	ADB0 ON	ADB1 CK	ADB0 CK
0	0	0	0	0	1	0	1

(4) Jump into the subroutine 2 developed on the internal RAM in advance. Add 6 to the value of the stack pointer (AR3 register) to remove the program counter (PC) value and PSW value stored by the $\overline{\text{NMI}}$ interrupt from the stack. In addition, clear the NMIF flag of the NMICR register to 0.

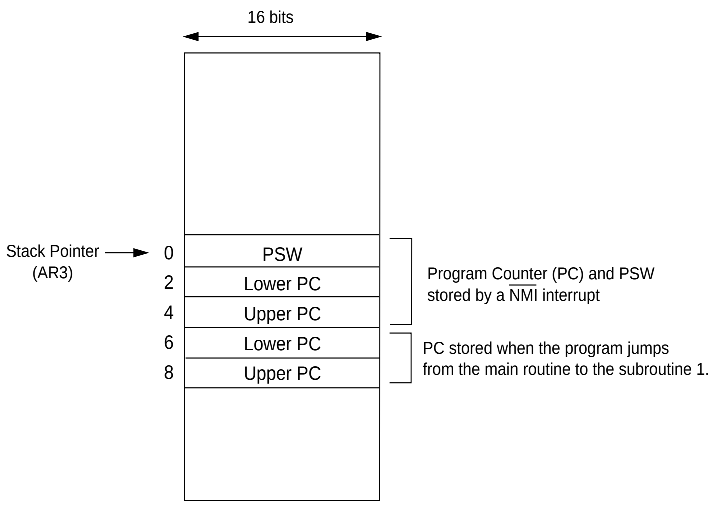


Figure 9-1-4 Stack State after NMI Interrupt

(5) Execute the subroutine 2 on the internal RAM and return to the original main routine with the RTS instruction.

9-2 System Related Register Protection

9-2-1 Overview

The MN102H55D/55G/F55G contains the system control register which prohibits programming the system related registers by the erroneous operations. Writing the value except x'7D' to the system control register prohibits programming the system related registers.

9-2-2 Control Registers

SYSCTL: x'00FCD0'

	7	6	5	4	3	2	1	0	
	SYS C7	SYS C6	SYS C5	SYS C4	SYS C3	SYS C2	SYS C1	SYS C0	System Control Register
At reset:	0	1	1	1	1	1	0	1	

7D: Program all registers (at reset)
Others: Do not program the following the system related registers

CPU Control	: CPUM, EFCR
Address Break	: ADB0, ADB1, ADBCTL
Memory Control	: EXWMD, MEMMD1, MEMMD2 DRAMMD1, DRAMMD2
Port Control	: P0MD, P1LMD, P1HMD
	: P2MD, P3LMD, P3HMD
	: P4LMD, P4HMD, P5LMD
	: P5HMD, P6MD, P7LMD
	: P7HMD, P8LMD, P8MMD
	: P8HMD, P9LMD, P9HMD
	: PAMD, PBMD

Chapter 10 Low-power Modes

10

10-1 Summary of Low-power Modes

10-1-1 Overview

The MN102H55D/55G/F55G provides two oscillation pins (high-speed and low speed) for system clock. It has two CPU operating modes, NORMAL and SLOW, and two standby modes, HALT and STOP. Using these modes effectively helps to reduce power consumption.

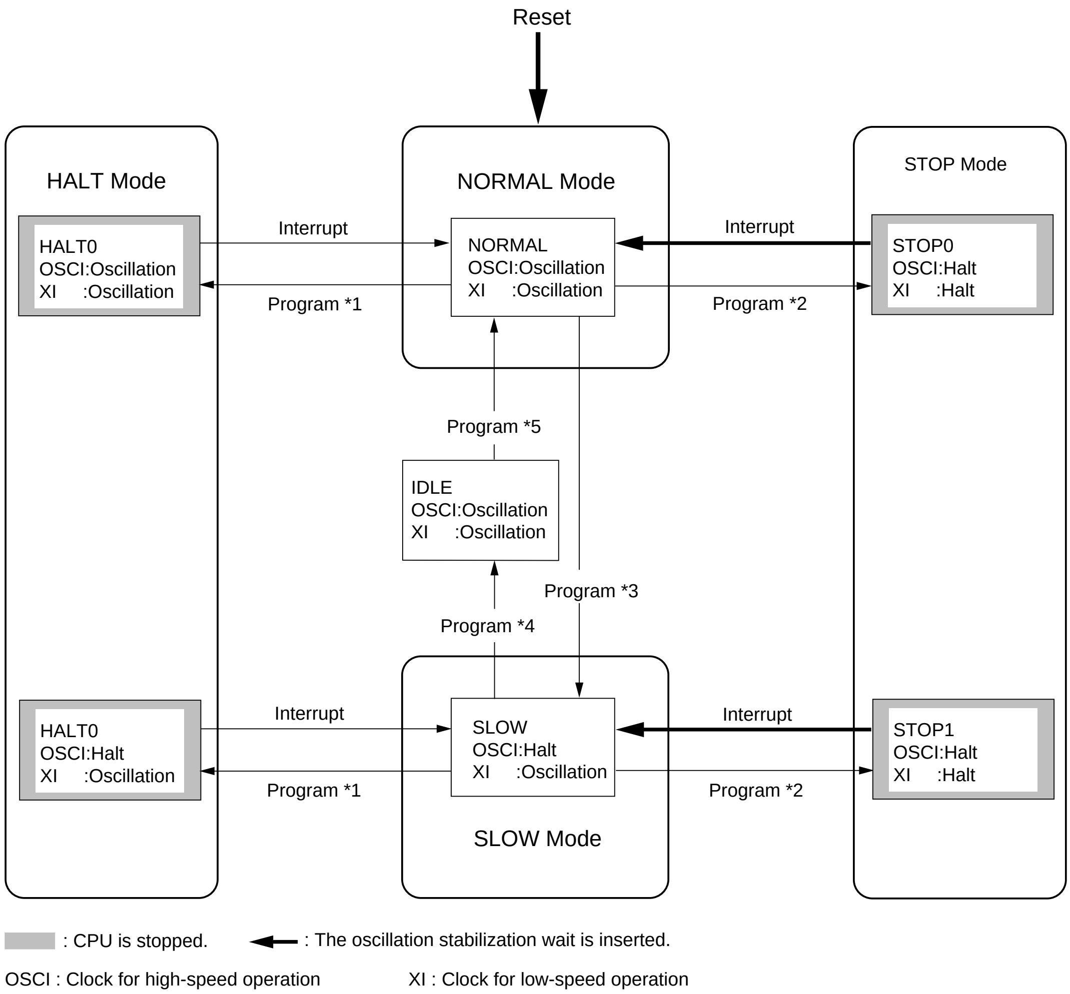


Figure 10-1-1 CPU Operating Mode Changes

Sample programs for program *1 to program *5 are described on the following pages.

The MN102H55D/55G/F55G contains two oscillation circuits for system clock. OSCI is the pin for high-speed operation (in NORMAL mode) while XI is the pin for low-speed operation (in SLOW mode). The CPU mode control register (CPUM) controls the transitions between NORMAL mode and SLOW mode or from NORMAL/ SLOW mode to standby mode. A normal reset or an interrupt recovers the CPU from standby mode. The oscillation stabilization wait occurs when the CPU is reset or when the CPU returns from STOP mode. The oscillation stabilization wait does not occur when the CPU returns from HALT mode. when the CPU returns from standby mode, NORMAL/STOP mode becomes the state before the CPU enters the standby mode.

The current from pins and the input pin level must not be unstable to reduce power consumption in STOP mode or HALT mode. For output pins, either match the output level to the level input to this pin externally or set the pin to input. For input pins, high or low level is fixed externally.

10-1-2 Control Registers

The CPU mode control register (CPUM) controls each mode transition.

CPUM: x'00FC00'

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	WD RST	WDM 1	WDM 0	0	0	0	0	0	0	0	0	OSCID	STOP	HALT	OSC1	OSC0
Reset :	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

System clock monitor flag

Operating Mode	STOP	HALT	OSC1	OSC0	OSCI/ OSCO	XI/XO	System Clock	CPU
NORMAL	0	0	0	0	Oscillation	Oscillation	OSCI/OSCO	Operation
IDLE	0	0	0	1	Oscillation	Oscillation	XI/XO	Operation
SLOW	0	0	1	1	Stop	Oscillation	XI/XO	Operation
HALT0 (*1)	0	1	0	0	Oscillation	Oscillation	OSCI/OSCO	Stop
HALT1 (*1)	0	1	1	1	Stop	Oscillation	XI/XO	Stop
STOP0 (*2)	1	0	0	0	Stop	Stop	Stop	Stop
STOP1 (*2)	1	0	1	1	Stop	Stop	Stop	Stop

*1 The CPU transits from NORMAL mode to HALT0 mode and from SLOW mode to HALT1 mode.
*2 The CPU transits from NORMAL mode to STOP0 mode and from SLOW mode to STOP1 mode.

Figure 10-1-2 Operating Mode Control and Clock Oscillation On/Off

Table 10-1-1 Watchdog Interrupt Interval

WDM1	WDM0	Watchdog Interrupt Interval (BOSC Signal)	Expected Return Time from STOP
0	0	2^{17} Cycles	$2^{17} \times$ (1/fosc1)
0	1	2^{5} Cycles	$2^{5} \times$ (1/fosc1)
1	0	2^{13} Cycles	$2^{13} \times$ (1/fosc1)
1	1	2^{15} Cycles	$2^{15} \times$ (1/fosc1)

The program changes the oscillation stabilization wait time. The oscillation stablization wait time is 2^{17} at reset. The time for error detection function changes as a result of the oscillation stablization wait time change.

Set only '00' to the WDM flags in the MN102HF55G.

WDM bits reduce the oscillation stabilization wait time from STOP mode. When both WDM1 and WDM0 are 0, the expected oscillation stabilization time is calculated as follows.

Oscillation stabilization wait time (tosciw) of the oscillation frequency fosci is:

$$tosciw = 2^{17} \times (1/fosci)$$

For example, tosciw = 3.85506 ms when fosci is 34 MHz.

10-1-3 Transferring between SLOW Mode and NORMAL Mode

The MN102H55D/55G/F55G has two CPU operating modes, NORMAL and SLOW. The CPU needs to go through IDLE mode when switching from SLOW mode to NORMAL mode.

The system clock monitor flag (OSCID) finds out whether the system clock for the existing peripheral function is the high-speed oscillation clock or the low-speed oscillation clock. Setting OSCID to 0 means that the high-speed oscillation clock is selected while setting OSCID to 1 means that the low-speed oscillation clock is selected.

■ Transferring from NORMAL mode to SLOW mode

The CPU can switch from NORMAL mode to SLOW mode by setting only the CPU mode control register because the low-speed oscillation clock operates stably. In this case, the CPU does not need to go through IDLE mode. The following is the program example of switching from NORMAL mode to SLOW mode.

```
Program *3
        mov   0xfc00, a1
        mov   (a1), d0      ; Read CPUM register
        or    0x3, d0       ; Set SLOW mode
        mov   d0, (a1)
```

The CPU operates based on low-speed oscillation clock in IDLE mode.

■ Transferring from SLOW mode to NORMAL mode

When the CPU transits from SLOW mode to NORMAL mode, the CPU needs to wait in IDLE mode with the program until the high-speed oscillation clock starts oscillation and becomes stable (It takes at least 3.9 ms when the CPU switches from SLOW mode to NORMAL mode (at 34 MHz operation)). The following is the program example of switching from SLOW mode to NORMAL mode.

```
Program *4
            mov   0xfc00, a1
            mov   (a1), d0      ; Read CPUM register
            and   0xfffd, d0    ; Set IDLE mode
            mov   d0, (a1)

Program *5
            mov   21, d0        ; This is the loop of waiting for 3.9 ms
   LOOP     add   -1, d0        ; in 32-kHz clock operation to switch
            bne   LOOP          ; from 32-kHz clock operation to 34-
                                ; MHz clock operation.

            mov   0xfc00, a1    ; Not required when the program
                                ; continues from program *4
            mov   (a1), d0      ; Read CPUM register
            and   0xfff0, d0    ; Set NORMAL mode
            mov   d0, (a1)
```

The oscillation stabilization wait time is required to stabilize oscillation. The program needs to count the same time as the oscillation stabilization time.

10-1-4 Switching to Standby Mode

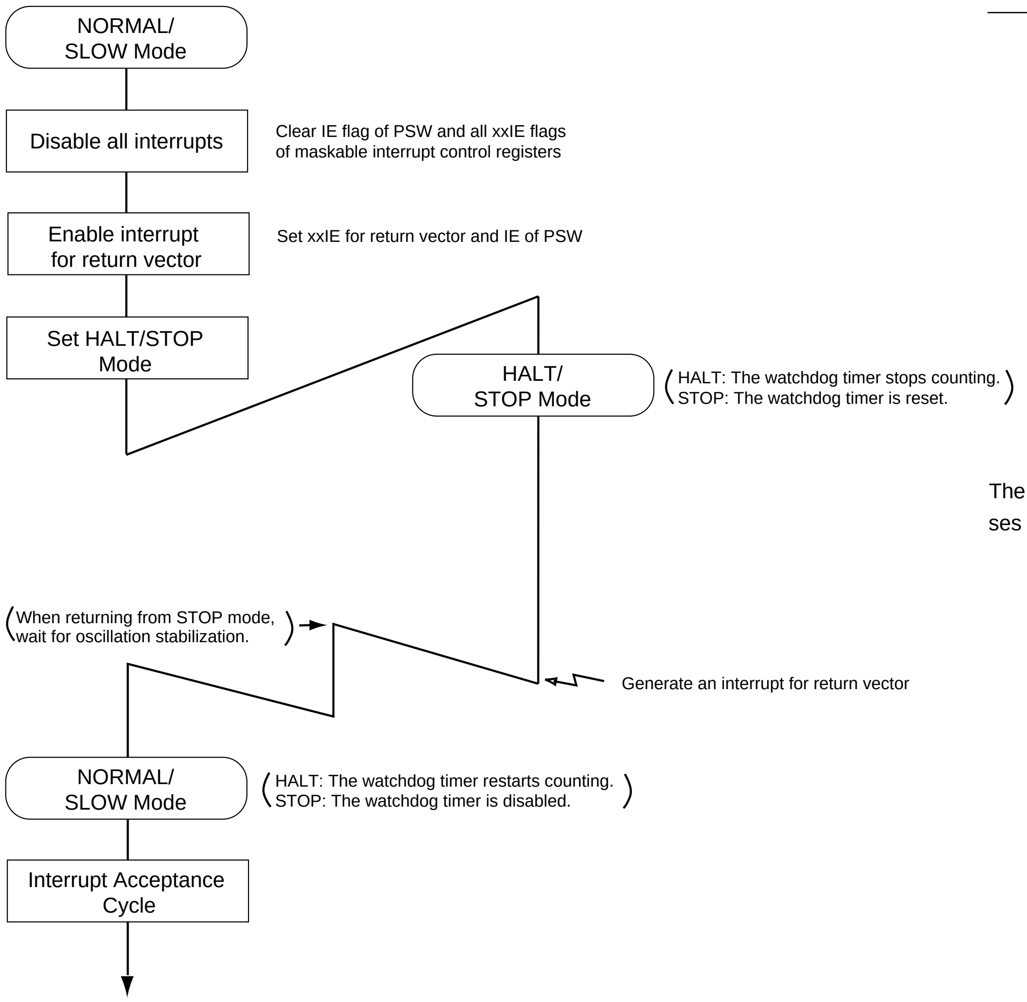
The program transits the CPU from the CPU operating mode to the standby mode. An interrupt switches the CPU from the standby mode to the CPU operating mode.

The following procedures are required before transferring to the standby mode.

(1) Clear the interrupt enable flag (IE) of the processor status word (PSW) and the interrupt enable flag (xxIE) of the maskable interrupt control register (xxICH) to disable all interrupts temporarily.

(2) Specify interrupt vector for returning from standby mode to the CPU operating mode and set only appropriate xxIE. In addition, set the IE flag of PSW.

The CPU cannot recover to the CPU operating mode when interrupt is enabled and the interrupt priority level is higher than the mask level set in PSW before switching to the CPU operating mode.



The process shown in parentheses is controlled by hardware.

Figure 10-1-3 Sequence of Switching to/from Standby Mode

■ Switching to HALT mode

The CPU switches from NORMAL mode to HALT0 mode and from SLOW mode to HALT1 mode. In both cases, only CPU stops keeping oscillation status. When the CPU switches to HALT mode while the watchdog timer is enabled, the watchdog timer stops counting. The following is the program example of switching to HALT mode.

Assign the JMP instruction and set the CPUM write to an even address with ALIGN instruction. This prevents the effects due to the difference of memory mode and expansion bus widths and outcomes the same result under any conditions.

Note:
The ALIGN value must be set to more than 2 when the ALIGN value is set by the quasi-SECTION instruction before this example within the file describing the program.

```
Program *1
            mov    (a1), d0     ; Read CPUM
            or     0x4, d0      ; Set HALT mode
            jump   stp_hlt      ; Branch unconditionally to an even ad-
            align  2            ; dress to erase the difference of operating
                                ; conditions.
stp_hlt     mov    d0, (a1)
            nop                 ; Insert more than three nops to execute a
            nop                 ; few instructions in the state of pipeline
            nop                 ; after writing to CPUM.
```

■ Returning from HALT mode

An interrupt or a reset recovers the CPU from HALT mode. Reset proceeds normal operation. An interrupt returns the previous mode before entering HALT mode and the watchdog timer restarts counting.

■ Switching to STOP mode

The CPU transits from NORMAL mode to STOP0 mode and from SLOW mode to STOP1 mode. In both cases, the oscillation and the CPU stop. When the CPU switches to STOP mode, the watchdog timer is reset. The following is the program example of switching to STOP mode.

When the CPU ends the oscillation stabilization wait and switches to the CPU operating mode, the watchdog timer becomes disabled automatically. When the watchdog timer operation is required, set the watchdog timer enabled.

```
Program *2
            mov    0xfc00, a1
            mov    (a1), d0     ; Read CPUM
            or     0x8, d0      ; Set STOP mode
            jump   stp_hlt      ; Branch unconditionally to an even ad-
            align  2            ; dress to erase the difference of operating
                                ; conditions.
stp_hlt     mov    d0, (a1)
            nop                 ; Insert more than three nops to execute a
            nop                 ; few instructions in the state of pipeline
            nop                 ; after writing to CPUM.
```

The oscillation stabilization wait is executed by hardware when returning from STOP mode. The program does not need to count the oscillation stabilization wait time.

■ Returning from STOP mode

An interrupt or a reset recovers the CPU from STOP mode. At reset, the watchdog timer becomes disabled after operating as the oscillation stabilization wait counter.

Chapter 11 Appendix

11

11-1 Electrical Characteristics

11-1-1 MN102H55D/55G

!

This LSI user's manual describes standard specifications. When using this LSI chip, please contact one of our sales offices for product standards.

Structure	CMOS integrated circuit
Application	General purpose
Function	16-bit microcontroller
Pin Configuration	Figure 1-4-1 to Figure 1-4-9
External Dimensions	Figure 1-4-14

A. Absolute Maximum Ratings

$V_{SS} = 0$ V

Parameter		Symbol	Rating	Unit
A1	Power supply voltage	V_{DD}	- 0.3 to + 4.6	V
A2	Input pin voltage	V_I	- 0.3 to V_{DD} + 0.3	V
A3	Output pin voltage	V_O	- 0.3 to V_{DD} + 0.3	V
A4	Input/output pin voltage	V_{IO}	- 0.3 to V_{DD} + 0.3	V
A5	Operating ambient temperature	Topr	- 40 to + 85	℃
A6	Storage temperature	Tstg	-55 to + 125	℃

Note:

1. Absolute Maximum Ratings are stress ratings not to cause damage to the device.
   Operation at these ratings is not guaranteed.
2. All of the V_{DD} and Vss pins are external pins. Connect them directly to the power source and ground.
3. To prevent latch-up tolerance, connect more than one by-pass condenser between power supply pins and ground. Use at least 0.2 µF condenser.

B. Operating Conditions

V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
B1	Power supply voltage	V_{DD}		3.0	3.3	3.6	V
	Crystal Oscillator 1 (OSCI)						
B2	Oscillator frequency	F_{osc1}		4		34	MHz
	Crystal Oscillator 2 (XI)						
B3	Oscillator frequency	F_{osc2}		32		166	kHz

C. Electrical Characteristics

1. DC Characteristics

V_{DD} = 3.3 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
C1	Power supply current during operation	$I_{DD}1$	$V_I = V_{DD}$ or V_{SS} Fosc1 = 34 MHz Output pins open			50	mA
C2	Power supply current in SLOW mode	$I_{DD}2$	$V_I = V_{DD}$ or V_{SS} Fosc2 = 32 kHz Output pins open			5	mA
C3	Power supply current in STOP mode	$I_{DD}3$	Oscillator stop All functions stop			70	µA
C4	Power supply current in HALT0 mode	$I_{DD}4$	Fosc1 = 34 MHz Fosc2 = 32 kHz			23	mA
C5	Power supply current in HALT1 mode	$I_{DD}5$	Fosc1 = oscillator stop Fosc2 = 32 kHz			1	mA

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
Input/Output Pins 1 < Output pushpull/Input LVTTL level schmidt trigger/Programmable pullup > P00-P07, P10-P17, P20-P27, P30-P37, P40-P43, P50-P57 P60-P63, P70-P75, P82-P87, P90-P93, PA0-PA5, PB0(BOSC)							
C6	Input high voltage	V_{IH1}		2.2			V
C7	Input low voltage	V_{IL1}				0.6	V
C8	Output high voltage	V_{OH1}	I_{OH} = -2.0 mA V_{DD} =3.3 V	2.4			V
C9	Output low voltage	V_{OL1}	I_{OL} = 2.0 mA V_{DD} =3.3 V			0.4	V
C10	Output leakage current	I_{LO1}	Vo = Hi-z	-10		10	μA
C11	Pullup resistance	P_{PU1}	V_I = V_{SS} V_{DD} =3.3 V	10	30	90	kΩ

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
Input/Output Pins 2 < Output pushpull/Input CMOS level schmidt trigger/Programmable pullup/Analog pins > P44-P47(AN4-7), P80-P81(DAC0-1), P94-P97(AN0-3)							
C12	Input high voltage	V_{IH2}		V_{DD}×0.8			V
C13	Input low voltage	V_{IL2}				V_{DD}×0.2	V
C14	Output high voltage	V_{OH2}	I_{OH} = -2.0 mA V_{DD} = 3.3 V	V_{DD}-0.6			V
C15	Output low voltage	V_{OL2}	I_{OL} = 2.0 mA V_{DD} = 3.3 V			0.4	V
C16	Output leakage current	I_{LO2}	Vo = Hi-Z	-10		10	µA
C17	Pullup resistance	P_{PU2}	V_I = V_{SS} V_{DD} = 3.3 V	10	30	90	kΩ

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
Input Pins < Input CMOS level schmidt trigger > /NMI, MODE, /RST							
C18	Input high voltage	V_{IH3}		V_{DD}×0.9			V
C19	Input low voltage	V_{IL3}				V_{DD}×0.1	V
C20	Input leakage current	V_{OH3}	V_{DD}=3.6 V V_I=V_{SS} to V_{DD}	-10		10	µA
OSCI pin, XI pin (at external clock input) : crystal, ceramic self-excited oscillation See Figure 1-4-10 to Figure 1-4-11							
C21	Input high voltage	V_{IH4}		V_{DD}×0.8		V_{DD}	V
C22	Input low voltage	V_{IL4}		V_{SS}		V_{DD}×0.2	V
Pin Capacitance							
C23	Input pin	C_{IN}	Ta=25 °C		7	15	pF
C24	Output pin	C_{OUT}			7	15	pF
C25	Input/output pin	$C_{I/O}$			7	15	pF

D. A/D Converter Characteristics

$V_{DD} = AV_{DD} = 3.3$ V
$V_{SS} = AV_{SS} = 0$ V
Ta = 25 °C

	Parameter	Symbol	Conditions	Capacitance Min	Capacitance Typ	Capacitance Max	Unit
D1	Resolution	L_{SB1}				10	Bits
D2	Zero-scale transition voltage	V_{ZS}	V_{REF+}=3.3 V V_{REF-}=0 V	-16.1		16.1	mV
D3	Full-scale transition voltage	V_{FS1}	V_{REF+}=3.3 V V_{REF-}=0 V	3.28		3.32	V
D4	Non-linearity error	N_{LE1}	V_{REF+}=3.3 V V_{REF-}=0 V	-4		4	LSB
D5	Differential non-linearity error	D_{NLE1}	V_{REF+}=3.3 V V_{REF-}=0 V	-4		4	LSB
D6	A/D conversion time	T_{SET1}	Fosc=34 MHz At 10-bit resolution	3.29			µs
D7	A/D conversion cycle	T_{SET2}	Fosc=34 MHz At 10-bit resolution	3.29			µs
D8	Analog input voltage	V_{IA}		V_{REF-}		V_{REF+}	V

Note: 1. Always set in relation of VDD >= AVDD >= VREF+ > VREF- >= AVSS >= VSS.

E. D/A Converter Characteristics

$V_{DD} = AV_{DD} = 3.3$ V
$V_{SS} = AV_{SS} = 0$ V
Ta = 25 °C

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
E1	Resolution	L_{SB2}				8	Bits
E2	Non-linearity error	N_{LE2}	V_{REF}+=3.3 V V_{REF}-=0 V		±2.0	±3.0	LSB
E3	Differential non-linearity error	D_{NLE2}	V_{REF}+=3.3 V V_{REF}-=0 V		±2.5	±3.5	LSB
E4	Settling time	T_{SET3}	CL=70 pF		3	6	µs
E5	Reference voltage	V_{REF}+		2.0		V_{DD}	V
		V_{REF}-		Vss		1.0	V
E6	Reference voltage pin input leakage current	I_{LO7}		-10		10	µA
E7	Analog output resistance	R_{OUT}		3	12	20	kΩ

Note: The capacitance values of E2, E3 are operational under $V_{DD} = V_{REF}+ = 3.3$ V, $V_{SS} = V_{REF}- = 0$ V.

F. AC Characteristics

Input Timing Conditions

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
External Clock Input Timing (Fosc1 = 34 MHz)							
F1	External clock input cycle time	t_{EXCcyc}	Fig 11-1-1	29.4			ns
F2	external clock input high pulse width	t_{EXCH}		$\frac{t_{EXCcyc}}{2}$ -3			ns
F3	External clock input low pulse width	t_{EXCL}		$\frac{t_{EXCcyc}}{2}$ -3			ns
F4	External clock input rise time	t_{EXCR}				3	ns
F5	External clock input fall time	t_{EXCF}				3	ns
Reset Input Timing							
F6	Reset signal pulse width (/RST)	t_{RSTW}	Fig 11-1-2	4			t_{EXCcyc}
Power Rise Timing							
F7	V_{DD}-V_{PP} setup time	t_{VDP}	Fig 11-1-3	2 (Note)			ms

Note: V_{DD}-V_{PP} setup time (t_{VDP}) is the capacitance only for MN102HF55G.

Input Timing Conditions

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C

	Parameter	Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
Data Transfer Signal Input Timing							
F8	Data acknowledge signal setup time (WAIT)	t_{WS}	Fig 11-1-5 Fig 11-1-9	12			ns
F9	Data acknowledge signal hold time (WAIT)	t_{WH}		0			ns
Data Transfer Signal Input Timing							
F10	Read data setup time (D15-00)	t_{RDS}	Fig 11-1-4 Fig 11-1-5 FIg 11-1-6	25+tcyc $\times S_{RE}$*			ns
F11	Read data hold time (D15-00)	t_{RDH}	Fig 11-1-8 Fig 11-1-9 Fig 11-1-10	-tcyc $\times S_{RE}$*			ns
Bus Authority Request Input Timing							
F12	Bus authority request signal setup time (/BREQ)	t_{BREQS}	Fig 11-1-12	0			ns
F13	Bus authority request signal hold time (/BREQ)	t_{BREQH}		0			ns
Interrupt Signal Input Timing							
F14	Nonmaskable interrupt signal pulse width (NMI)	t_{NMIW}	Fig 11-1-13	10 (Note)			t_{cyc}
F15	External interrupt signal pulse width (/IRQ4~0)	t_{IRQW}		4 (Note)			t_{cyc}

Note : An interrupt may occur when the noise of the specified time or less is input.
* S_{RE} means /RE short mode. (S_{RE}=0, 0.5, 1, 1.5)

Input Timing Conditions

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
Serial Interface Related Signal Timing (Synchronous Serial Reception)							
F16	Data reception setup time (SBI4-0)	t_{RXDS}	FIg 11-1-17	17			ns
F17	Data reception hold time (SBI4-0)	t_{RXDH}		17			ns
F18	Transfer clock input high pulse width (SBT4-0)	t_{SCH}	Fig 11-1-16 Fig 11-1-17	tcyc×4			ns
F19	Transfer clock input low pulse width (SBT4-0)	t_{SCL}		tcyc×4			ns
Timer/Counter Signal Input Timing							
F20	Timer external input clock low pulse width (TMnIO: n=0, 4, 7) (TMnIOA, TMnIOB, TMnIC: n=8-12) (TMnIA, TMnIB: n=13-15)	t_{TCCLKL}	Fig 11-1-18	2			t_{cyc}
F21	Timer external input clock high pulse width (TMnIO: n=0, 4, 7) (TMnIOA, TMnIOB, TMnIC: n=8-12) (TMnIA, TMnIB: n=13-15)	t_{TCCLKH}		2			t_{cyc}

G. AC Characteristics (Output)

Output Signal Characteristics

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C
C_L = 70 pF

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
System Clock Output Timing							
G1	System clock output cycle time (BOSC)	t_{cyc}	Fig 11-1-1 Fig 11-1-4 to 11	33.3			ns
G2	System clock output low pulse width (BOSC)	t_{CL}		11.65			ns
G3	System clock output high pulse width (BOSC)	t_{CH}		11.65			ns
G4	System clock output rise time (BOSC)	t_{CR}				5	ns
G5	System clock output fall time (BOSC)	t_{CF}				5	ns

Output Signal Characteristics

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C
C_L = 70 pF

	Parameter	Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
Data Transfer Signal Output Timing 1							
G6	Address delay time 1 (A23-0), (A23-16), (AD15-0)	t_{AD1}	Fig 11-1-4 Fig 11-1-5 Fig 11-1-8 to 11			5	ns
G7	Address hold time 1 (A23-0), (A23-16)	t_{AH1}	Fig 11-1-4 Fig 11-1-8	tcyc×S*			ns
G8	Address hold time 2 (AD15-0)	t_{AH2}	ALE late 0, long 0 mode AD long 1 mode Fig 11-1-8	20			ns
			Other Modes Fig 11-1-8	tcyc×(LAD-LALE-1)*			
G9	Write data delay time (D15-0), (AD15-0)	t_{DD1}	Fig 11-1-4 to 5 Fig 11-1-7			5	ns
G10	Write data hold time (D15-0)	t_{DH1}	Fig 11-1-4 to 5 Fig 11-1-7 to 10	tcyc×S*			ns

* S means /WE short mode. (S=0,0.5,1,1.5)
LAD means AD long mode. (LAD=1,2,3)
LALE means ALE long mode. (LALE=0,0.5,1,1.5)

Output Signal Characteristics

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C
C_L = 70 pF

	Parameter	Symbol	Conditions	Capacitance Min	Capacitance Typ	Capacitance Max	Unit
Data Transfer Signal Output Timing 2							
G11	Chip-select signal fall delay time (/CS3-0), (/CS3-1)	t_{CSDF1}	Fig 11-1-4 to 5 Fig 11-1-8 to 9			6	ns
G12	Chip-select signal rise delay time (/CS3-0), (/CS3-1)	t_{CSDR1}				9	ns
G13	Chip-select signal hold time (/CS3-0)	t_{CSH}	Fig 11-1-4	t_{cyc}×S*			ns
G14	Address latch signal rise delay time (ALE)	t_{ALER1}	Fig 11-1-8 to 11			10	ns
G15	Address latch signal fall delay time (ALE)	t_{ALEF1}	Fig 11-1-8 to 11			5	ns

* S means /WE short mode. (S=0,0.5,1,1.5)

Output Signal Characteristics

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C
C_L = 70 pF

	Parameter	Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
Data Transfer Signal Output Timing 3							
G16	Read enable signal fall delay time 1 (/RE)	t_{REDF1}	Late 0.5 mode Fig 11-1-4 to 6			25	ns
			Other modes Fig 11-1-4 to 6			10	ns
G17	Read enable signal fall delay time 2 (/RE)	t_{REDF2}	ALE late 0, long 0 mode AD long 1 mode Fig 11-1-8 to 9			20	ns
			Other modes Fig 11-1-8 to 9			10	
G18	Read enable signal rise delay time (/RE)	t_{REDR1}	Fig 11-1-4 to 6			10	ns
G19	Write enable signal fall delay time 1 (/WEH, WEL)	t_{WEDF1}	Late 1 mode Fig 11-1-4 to 5			20	ns
			Other modes Fig 11-1-4 to 5 Fig 11-1-7 to 9			8	
G20	Write enable pulse width time (/WEH, WEL)	t_{WEPW}	Late 1 mode Fig 11-1-4 to 5 Fig 11-1-7 to 9	tcyc × (2W-L-S+2) -20*			ns
			Other modes Fig 11-1-4 to 5 Fig 11-1-7 to 9	tcyc× (2W-L-S+2) -10*			

* W is the number of waits. (W=0,0.5,1,1.5,..7)
L means /WE late mode. (L=1,2,3)
S means /WE short mode. (S=0,0.5,1,1.5)

Output Signal Characteristics

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C
C_L = 70 pF

	Parameter	Symbol	Conditions		Capacitance			Unit
					Min	Typ	Max	
Serial Interface Signal Output Timing (Synchronous Serial Transmission)								
G21	Transfer data delay time (SBO4-0)	t_{TXDD}	Fig 11-1-14 Fig 11-1-15 Fig 11-1-16	Normal			15	ns
				I^2C			$t_{cyc} \times 4$	ns
G22	Transfer data hold time (transfer in progress) (SBO4-0)	t_{TXDH1}	Fig 11-1-14		0			ns
G23	Transfer data hold time (Transfer end timing at SBT input) (SBO4-0)	t_{TXDH2}	Fig 11-1-15		t_{cyc} (Note)			ns
G24	Transfer data hold time (Transfer end timing at SBT output) (SBO4-0)	t_{TXDH3}	Fig 11-1-16		$\frac{t_{SCH}+t_{SCL}}{2}$			ns

Note: Set SBO4-0 output hold time to BOSC cycle or more in SCnCTR (n=4-0) register.

11-1-2 MN102HF55G

Structure	CMOS integrated circuit
Application	General purpose
Function	16-bit microcontroller
Pin Configuration	Figure 1-4-1 to Figure 1-4-9
External Dimensions	Figure 1-4-14

A. Absolute Maximum Ratings

V_{SS} = 0 V

Parameter		Symbol	Rating	Unit
A1	Power supply voltage	V_{DD}	- 0.3 to + 4.6	V
A2	VPP power supply voltage	V_{PP}	- 0.3 to + 6.0	V
A3	Input pin voltage	V_I	- 0.3 to V_{DD} + 0.3	V
A4	Output pin voltage	V_O	- 0.3 to V_{DD} + 0.3	V
A5	Input/output pin voltage	V_{IO}	- 0.3 to V_{DD} + 0.3	V
A6	Operating ambient temperature	Topr	- 40 to + 85	℃
A7	Storage temperature	Tstg	-55 to + 125	℃

Note:

1. Absolute Maximum Ratings are stress ratings not to cause damage to the device. Operation at these ratings is not guaranteed.
2. All of the V_{DD} and Vss pins are external pins. Connect them directly to the power source and ground.
3. To prevent latch-up tolerance, connect more than one by-pass condenser between power supply pins and ground. Use at least 0.2 µF condenser.

B. Operating Conditions

V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
B1	Power supply voltage	V_{DD}		3.0	3.3	3.6	V
	Crystal Oscillator 1 (OSCI)						
B2	Oscillator frequency	F_{osc1}		4		30	MHz
	Crystal Oscillator 2 (XI)						
B3	Oscillator frequency	F_{osc2}		32		166	kHz

Operating Conditions for Flash EEPROM Version

V_{SS} = 0 V

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
B4	Power supply voltage	V_{DD}		3.0	3.3	3.6	V
B5	Vpp power supply voltage	V_{PP}		4.5	5.0	5.5	V
B6	Operating ambient temperature	Ta		0	25	50	℃
B7	Programming operations					10	times

C. Electrical Characteristics

1. DC Characteristics

V_{DD} = 3.3 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
C1	Power supply current during operation	$I_{DD}1$	$V_I = V_{DD}$ or V_{SS} Fosc1 = 30 MHz Output pins open			50	mA
C2	Power supply current in SLOW mode	$I_{DD}2$	$V_I = V_{DD}$ or V_{SS} Fosc2 = 32 kHz Output pins open			5	mA
C3	Power supply current in STOP mode	$I_{DD}3$	Oscillator stop All functions stop			70	µA
C4	Power supply current in HALT0 mode	$I_{DD}4$	Fosc1 = 30 MHz Fosc2 = 32 kHz			23	mA
C5	Power supply current in HALT1 mode	$I_{DD}5$	Fosc1 = oscillator stop Fosc2 = 32 kHz			1	mA

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
Input/Output Pins 1 < Output pushpull/Input LVTTL level schmidt trigger/Programmable pullup > P00-P07, P10-P17, P20-P27, P30-P37, P40-P43, P50-P57 P60-P63, P70-P75, P82-P87, P90-P93, PA0-PA5, PB0(BOSC)							
C6	Input high voltage	V_{IH1}		2.2			V
C7	Input low voltage	V_{IL1}				0.6	V
C8	Output high voltage	V_{OH1}	I_{OH} = -2.0 mA V_{DD} =3.3 V	2.4			V
C9	Output low voltage	V_{OL1}	I_{OL} = 2.0 mA V_{DD} =3.3 V			0.4	V
C10	Output leakage current	I_{LO1}	Vo = Hi-z	-10		10	µA
C11	Pullup resistance	P_{PU1}	V_I = V_{SS} V_{DD} =3.3 V	10	30	90	kΩ

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
Input/Output Pins 2 < Output pushpull/Input CMOS level schmidt trigger/Programmable pullup/Analog pins > P44-P47(AN4-7), P80-P81(DAC0-1), P94-P97(AN0-3)							
C12	Input high voltage	V_{IH2}		V_{DD}×0.8			V
C13	Input low voltage	V_{IL2}				V_{DD}×0.2	V
C14	Output high voltage	V_{OH2}	I_{OH} = -2.0 mA V_{DD} = 3.3 V	V_{DD}-0.6			V
C15	Output low voltage	V_{OL2}	I_{OL} = 2.0 mA V_{DD} = 3.3 V			0.4	V
C16	Output leakage current	I_{LO2}	Vo = Hi-Z	-10		10	µA
C17	Pullup resistance	P_{PU2}	V_I = V_{SS} V_{DD} = 3.3 V	10	30	90	kΩ

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
Input Pins < Input CMOS level schmidt trigger > /NMI, MODE, /RST							
C18	Input high voltage	V_{IH3}		V_{DD}×0.9			V
C19	Input low voltage	V_{IL3}				V_{DD}×0.1	V
C20	Input leakage current	V_{OH3}	V_{DD}=3.6 V V_I=V_{SS} to V_{DD}	-10		10	µA
OSCI pin, XI pin (at external clock input) : crystal, ceramic self-excited oscillation See Figure 1-4-10 to Figure 1-4-11							
C21	Input high voltage	V_{IH4}		V_{DD}×0.8		V_{DD}	V
C22	Input low voltage	V_{IL4}		V_{SS}		V_{DD}×0.2	V
Pin Capacitance							
C23	Input pin	C_{IN}	Ta=25 °C		7	15	pF
C24	Output pin	C_{OUT}			7	15	pF
C25	Input/output pin	$C_{I/O}$			7	15	pF

D. A/D Converter Characteristics

$V_{DD} = AV_{DD} = 3.3$ V
$V_{SS} = AV_{SS} = 0$ V
Ta = 25 ℃

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
D1	Resolution	L_{SB1}				10	Bits
D2	Zero-scale transition voltage	V_{ZS}	$V_{REF}+$=3.3 V $V_{REF}-$=0 V	-16.1		16.1	mV
D3	Full-scale transition voltage	V_{FS1}	$V_{REF}+$=3.3 V $V_{REF}-$=0 V	3.28		3.32	V
D4	Non-linearity error	N_{LE1}	$V_{REF}+$=3.3 V $V_{REF}-$=0 V	-5		5	LSB
D5	Differential non-linearity error	D_{NLE1}	$V_{REF}+$=3.3 V $V_{REF}-$=0 V	-5		5	LSB
D6	A/D conversion time	T_{SET1}	Fosc=30 MHz At 10-bit resolution	3.73			µs
D7	A/D conversion cycle	T_{SET2}	Fosc=30 MHz At 10-bit resolution	3.73			µs
D8	Analog input voltage	V_{IA}		$V_{REF}-$		$V_{REF}+$	V

Note: 1. Always set in relation of VDD >= AVDD >= VREF+ > VREF- >= AVSS >= VSS.

E. D/A Converter Characteristics

$V_{DD} = AV_{DD} = 3.3$ V
$V_{SS} = AV_{SS} = 0$ V
Ta = 25 °C

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
E1	Resolution	L_{SB2}				8	Bits
E2	Non-linearity error	N_{LE2}	V_{REF+}=3.3 V V_{REF-}=0 V		±2.0	±3.0	LSB
E3	Differential non-linearity error	D_{NLE2}	V_{REF+}=3.3 V V_{REF-}=0 V		±2.5	±3.5	LSB
E4	Settling time	T_{SET3}	CL=70 pF		3	6	µs
E5	Reference voltage	V_{REF+}		2.0		V_{DD}	V
		V_{REF-}		Vss		1.0	V
E6	Reference voltage pin input leakage current	I_{LO7}		-10		10	µA
E7	Analog output resistance	R_{OUT}		3	12	20	kΩ

Note: The capacitance values of E2, E3 are operational under $V_{DD} = V_{REF+} = 3.3$ V, $V_{SS} = V_{REF-} = 0$ V.

F. AC Characteristics

Input Timing Conditions

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C

	Parameter	Symbol	Conditions	Capacitance Min	Capacitance Typ	Capacitance Max	Unit
External Clock Input Timing (Fosc1 = 30 MHz)							
F1	External clock input cycle time	t_{EXCcyc}	Fig 11-1-1	33.3			ns
F2	external clock input high pulse width	t_{EXCH}		$\frac{t_{EXCcyc}}{2}$ -3			ns
F3	External clock input low pulse width	t_{EXCL}		$\frac{t_{EXCcyc}}{2}$ -3			ns
F4	External clock input rise time	t_{EXCR}				3	ns
F5	External clock input fall time	t_{EXCF}				3	ns
Reset Input Timing							
F6	Reset signal pulse width (/RST)	t_{RSTW}	Fig 11-1-2	4			t_{EXCcyc}
Power Rise Timing							
F7	V_{DD}-V_{PP} setup time	t_{VDP}	Fig 11-1-3	2 (Note)			ms

Note: V_{DD}-V_{PP} setup time (t_{VDP}) is the capacitance only for MN102HF55G.

Input Timing Conditions

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C

	Parameter	Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
Data Transfer Signal Input Timing							
F8	Data acknowledge signal setup time (WAIT)	t_{WS}	Fig 11-1-5 Fig 11-1-9	12			ns
F9	Data acknowledge signal hold time (WAIT)	t_{WH}		0			ns
Data Transfer Signal Input Timing							
F10	Read data setup time (D15-00)	t_{RDS}	Fig 11-1-4 Fig 11-1-5 FIg 11-1-6 Fig 11-1-8 Fig 11-1-9 Fig 11-1-10	25+tcyc ×S_{RE}*			ns
F11	Read data hold time (D15-00)	t_{RDH}		-tcyc ×S_{RE}*			ns
Bus Authority Request Input Timing							
F12	Bus authority request signal setup time (/BREQ)	t_{BREQS}	Fig 11-1-12	0			ns
F13	Bus authority request signal hold time (/BREQ)	t_{BREQH}		0			ns
Interrupt Signal Input Timing							
F14	Nonmaskable interrupt signal pulse width (NMI)	t_{NMIW}	Fig 11-1-13	10 (Note)			t_{cyc}
F15	External interrupt signal pulse width (/IRQ4~0)	t_{IRQW}		4 (Note)			t_{cyc}

Note : An interrupt may occur when the noise of the specified time or less is input.
* S_{RE} means /RE short mode. (S_{RE}=0, 0.5, 1, 1.5)

Input Timing Conditions

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
Serial Interface Related Signal Timing (Synchronous Serial Reception)							
F16	Data reception setup time (SBI4-0)	t_{RXDS}	FIg 11-1-17	17			ns
F17	Data reception hold time (SBI4-0)	t_{RXDH}		17			ns
F18	Transfer clock input high pulse width (SBT4-0)	t_{SCH}	Fig 11-1-16 Fig 11-1-17	tcyc×4			ns
F19	Transfer clock input low pulse width (SBT4-0)	t_{SCL}		tcyc×4			ns
Timer/Counter Signal Input Timing							
F20	Timer external input clock low pulse width (TMnIO: n=0, 4, 7) (TMnIOA, TMnIOB, TMnIC: n=8-12) (TMnIA, TMnIB: n=13-15)	t_{TCCLKL}	Fig 11-1-18	2			t_{cyc}
F21	Timer external input clock high pulse width (TMnIO: n=0, 4, 7) (TMnIOA, TMnIOB, TMnIC: n=8-12) (TMnIA, TMnIB: n=13-15)	t_{TCCLKH}		2			t_{cyc}

G. AC Characteristics (Output)

Output Signal Characteristics

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C
C_L = 70 pF

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
System Clock Output Timing							
G1	System clock output cycle time (BOSC)	t_{cyc}	Fig 11-1-1 Fig 11-1-4 to 11	33.3			ns
G2	System clock output low pulse width (BOSC)	t_{CL}		11.65			ns
G3	System clock output high pulse width (BOSC)	t_{CH}		11.65			ns
G4	System clock output rise time (BOSC)	t_{CR}				5	ns
G5	System clock output fall time (BOSC)	t_{CF}				5	ns

Output Signal Characteristics

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C
C_L = 70 pF

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
Data Transfer Signal Output Timing 1							
G6	Address delay time 1 (A23-0), (A23-16), (AD15-0)	t_{AD1}	Fig 11-1-4 Fig 11-1-5 Fig 11-1-8 to 11			5	ns
G7	Address hold time 1 (A23-0), (A23-16)	t_{AH1}	Fig 11-1-4 Fig 11-1-8	tcyc×S*			ns
G8	Address hold time 2 (AD15-0)	t_{AH2}	ALE late 0, long 0 mode AD long 1 mode Fig 11-1-8	20			ns
			Other Modes Fig 11-1-8	tcyc×(L_{AD}-L_{ALE}-1)*			
G9	Write data delay time (D15-0), (AD15-0)	t_{DD1}	Fig 11-1-4 to 5 Fig 11-1-7			5	ns
G10	Write data hold time (D15-0)	t_{DH1}	Fig 11-1-4 to 5 Fig 11-1-7 to 10	tcyc×S*			ns

* S means /WE short mode. (S=0,0.5,1,1.5)
L_{AD} means AD long mode. (L_{AD}=1,2,3)
L_{ALE} means ALE long mode. (L_{ALE}=0,0.5,1,1.5)

Output Signal Characteristics

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C
C_L = 70 pF

Parameter		Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
Data Transfer Signal Output Timing 2							
G11	Chip-select signal fall delay time (/CS3-0), (/CS3-1)	t_{CSDF1}	Fig 11-1-4 to 5 Fig 11-1-8 to 9			6	ns
G12	Chip-select signal rise delay time (/CS3-0), (/CS3-1)	t_{CSDR1}				9	ns
G13	Chip-select signal hold time (/CS3-0)	t_{CSH}	Fig 11-1-4	t_{cyc}×S*			ns
G14	Address latch signal rise delay time (ALE)	t_{ALER1}	Fig 11-1-8 to 11			10	ns
G15	Address latch signal fall delay time (ALE)	t_{ALEF1}	Fig 11-1-8 to 11			5	ns

* S means /WE short mode. (S=0,0.5,1,1.5)

Output Signal Characteristics

V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C
C_L = 70 pF

	Parameter	Symbol	Conditions	Capacitance			Unit
				Min	Typ	Max	
Data Transfer Signal Output Timing 3							
G16	Read enable signal fall delay time 1 (/RE)	t_{REDF1}	Late 0.5 mode Fig 11-1-4 to 6			25	ns
			Other modes Fig 11-1-4 to 6			10	ns
G17	Read enable signal fall delay time 2 (/RE)	t_{REDF2}	ALE late 0, long 0 mode AD long 1 mode Fig 11-1-8 to 9			20	ns
			Other modes Fig 11-1-8 to 9			10	
G18	Read enable signal rise delay time (/RE)	t_{REDR1}	Fig 11-1-4 to 6			10	ns
G19	Write enable signal fall delay time 1 (/WEH, WEL)	t_{WEDF1}	Late 1 mode Fig 11-1-4 to 5 Fig 11-1-7 to 9			20	ns
			Other modes Fig 11-1-4 to 5 Fig 11-1-7 to 9			8	
G20	Write enable pulse width time (/WEH, WEL)	t_{WEPW}	Late 1 mode Fig 11-1-4 to 5 Fig 11-1-7 to 9	t_{cyc} × (2W-L-S+2) -20*			ns
			Other modes Fig 11-1-4 to 5 Fig 11-1-7 to 9	t_{cyc}× (2W-L-S+2) -10*			

* W is the number of waits. (W=0,0.5,1,1.5,..7)
L means /WE late mode. (L=1,2,3)
S means /WE short mode. (S=0,0.5,1,1.5)

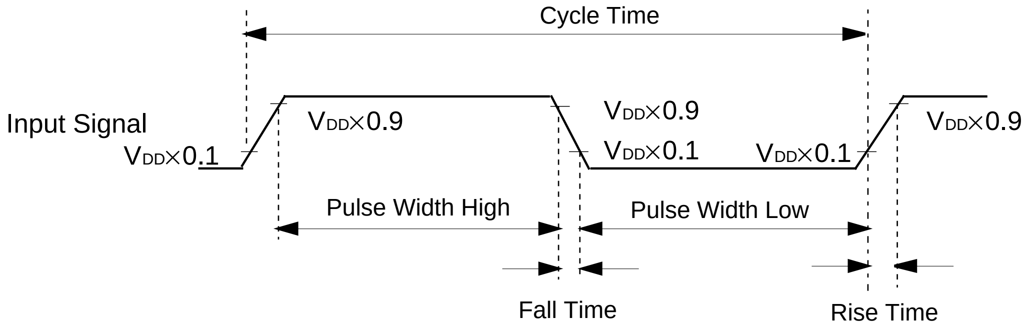
Output Signal Characteristics

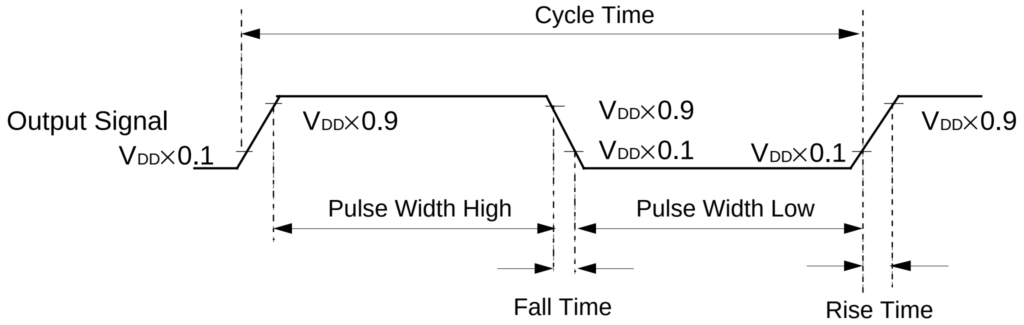
V_{DD} = 3.0 V to 3.6 V
V_{SS} = 0 V
Ta = -40 °C to +85 °C
C_L = 70 pF

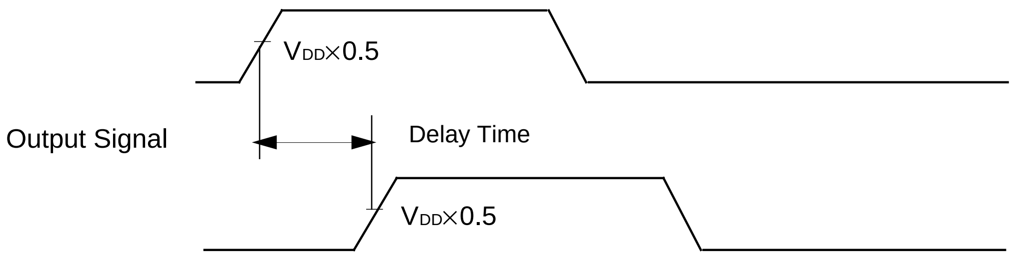
	Parameter	Symbol	Conditions		Capacitance Min	Capacitance Typ	Capacitance Max	Unit
Serial Interface Signal Output Timing (Synchronous Serial Transmission)								
G21	Transfer data delay time (SBO4-0)	t_{TXDD}	Fig 11-1-14 Fig 11-1-15 Fig 11-1-16	Normal			15	ns
				I²C			t_{cyc}×4	ns
G22	Transfer data hold time (transfer in progress) (SBO4-0)	t_{TXDH1}	Fig 11-1-14		0			ns
G23	Transfer data hold time (Transfer end timing at SBT input) (SBO4-0)	t_{TXDH2}	Fig 11-1-15		t_{cyc} (Note)			ns
G24	Transfer data hold time (Transfer end timing at SBT output) (SBO4-0)	t_{TXDH3}	Fig 11-1-16		$\frac{t_{SCH}+t_{SCL}}{2}$			ns

Note: Set SBO4-0 output hold time to BOSC cycle or more in SCnCTR (n=4-0) register.

AC Timing Voltage Level







(Both setup time and hold time are V_{DD}×0.5)

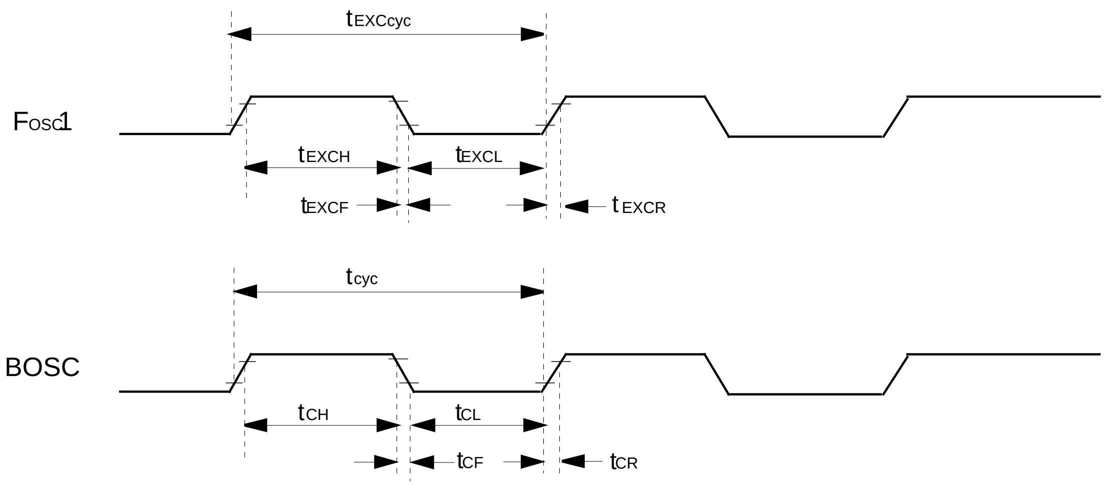


Figure 11-1-1 System Clock Timing

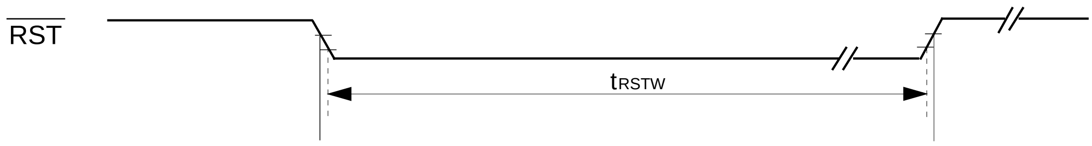


Figure 11-1-2 Reset Timing

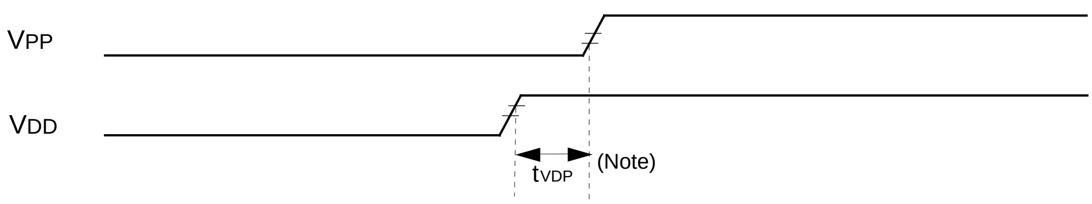


Figure 11-1-3 Voltage Rise Timing

Note: VDD-VPP setup time (t_{VDP}) is the capacitance value only for MN102HF55G.

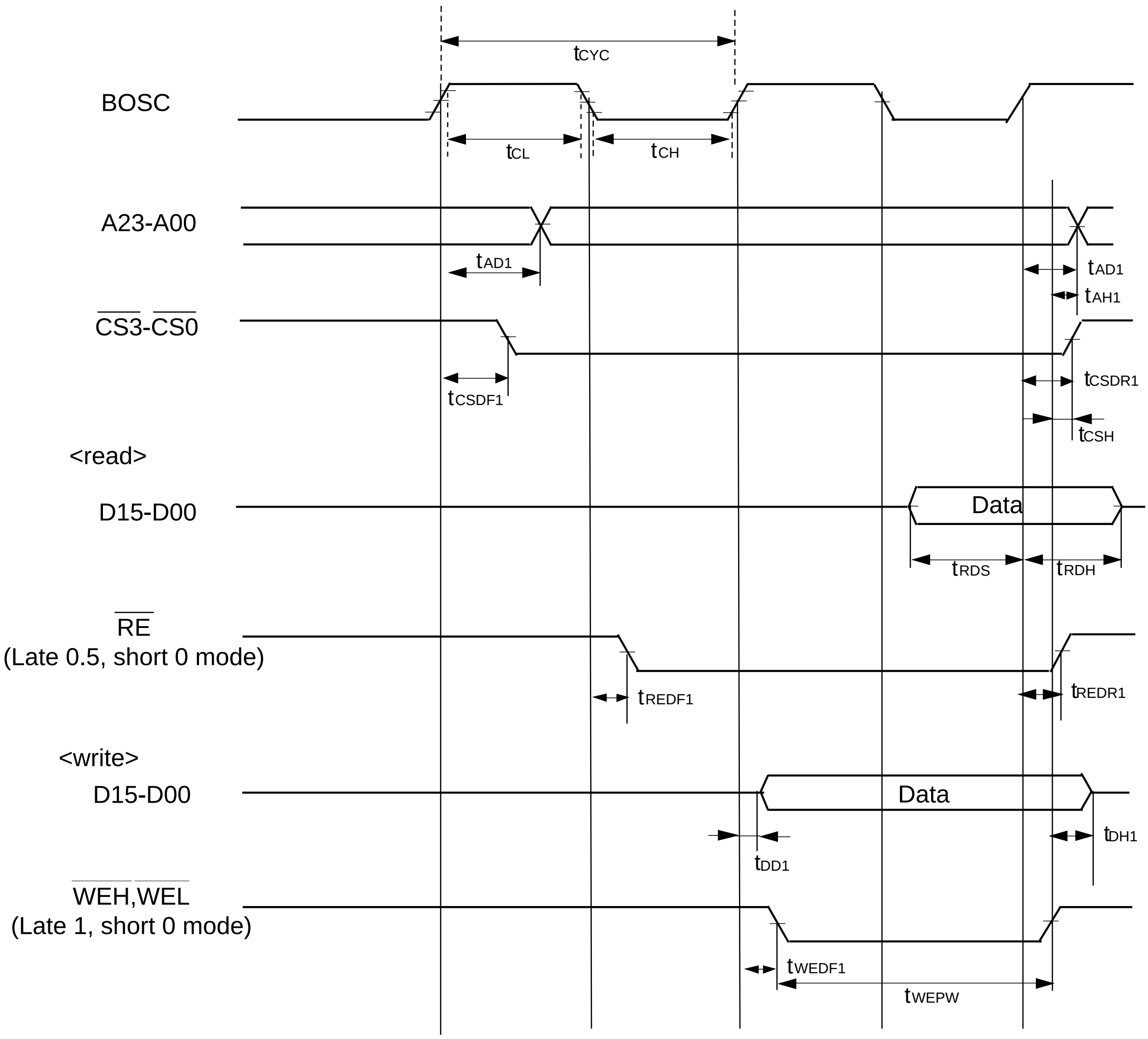


Figure 11-1-4 Data Transfer Signal Timing
(Address/Data Separate, Without Wait, Read/Write)

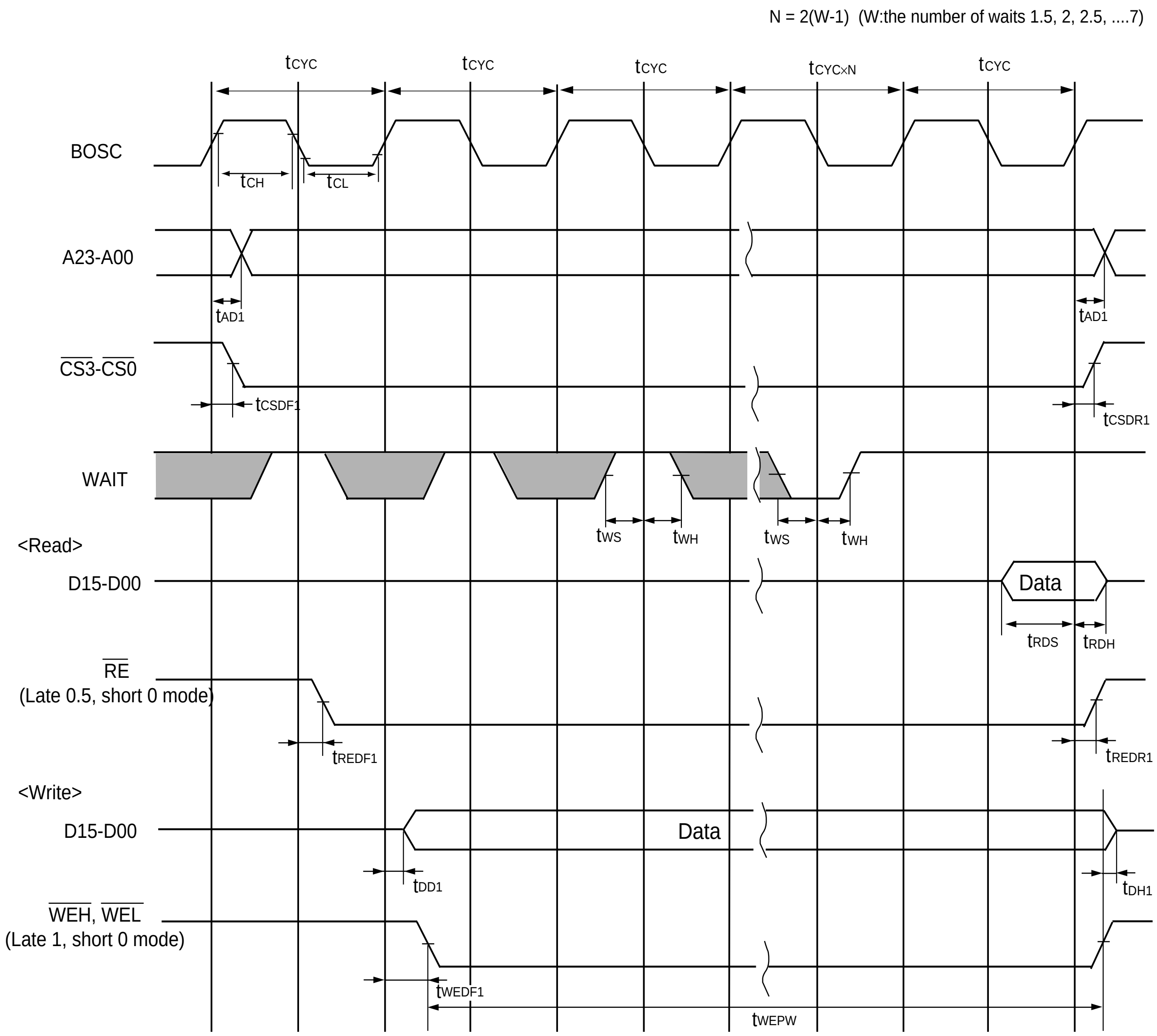


Figure 11-1-5 Data Transfer Signal Timing
(Address/Data Separate, With Wait (1.5 or More), Read/Write)

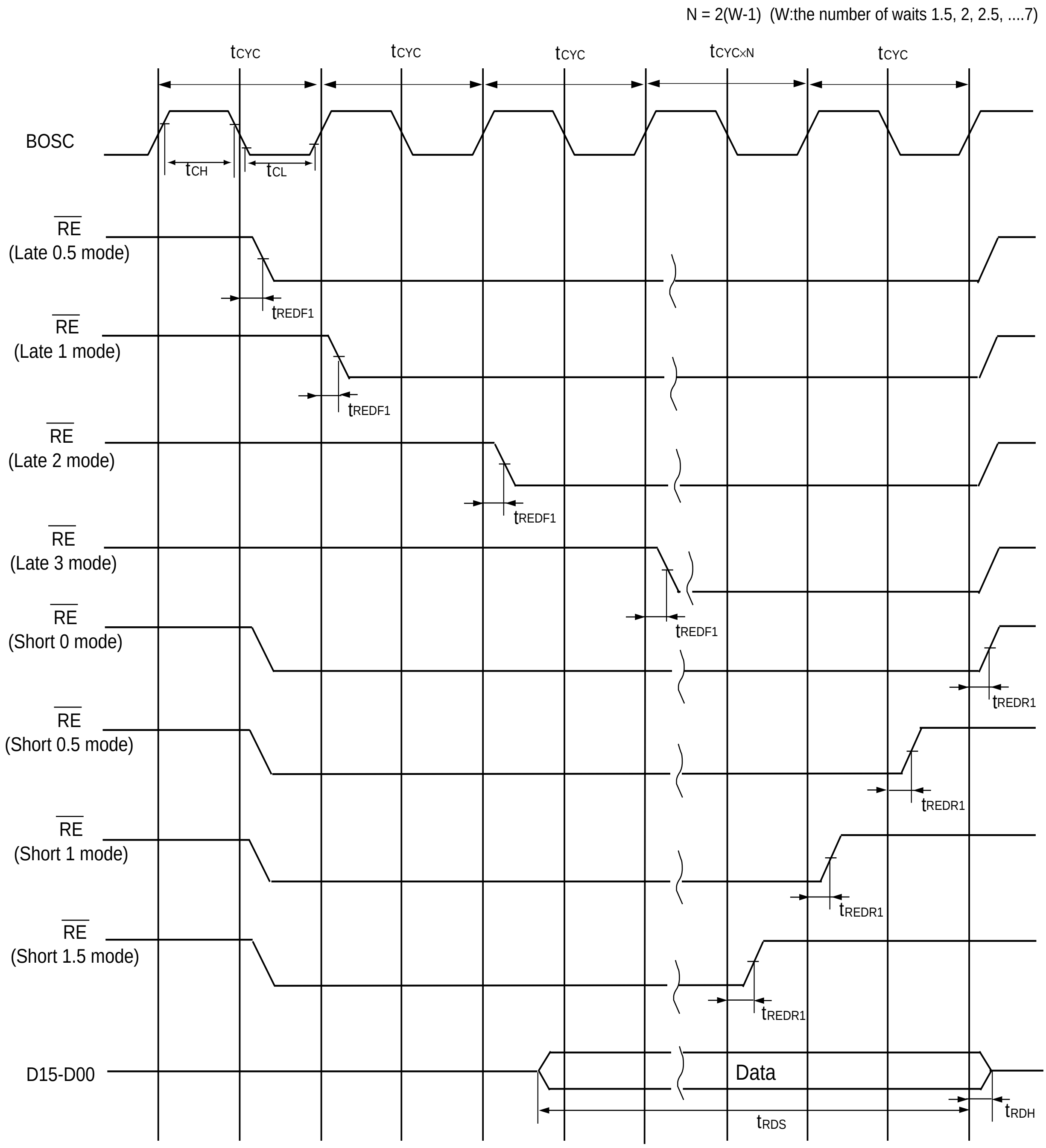


Figure 11-1-6 Data Transfer Signal Timing
(Address/Data Separate, With Wait (1.5 or More), /RE Late, Short Mode)

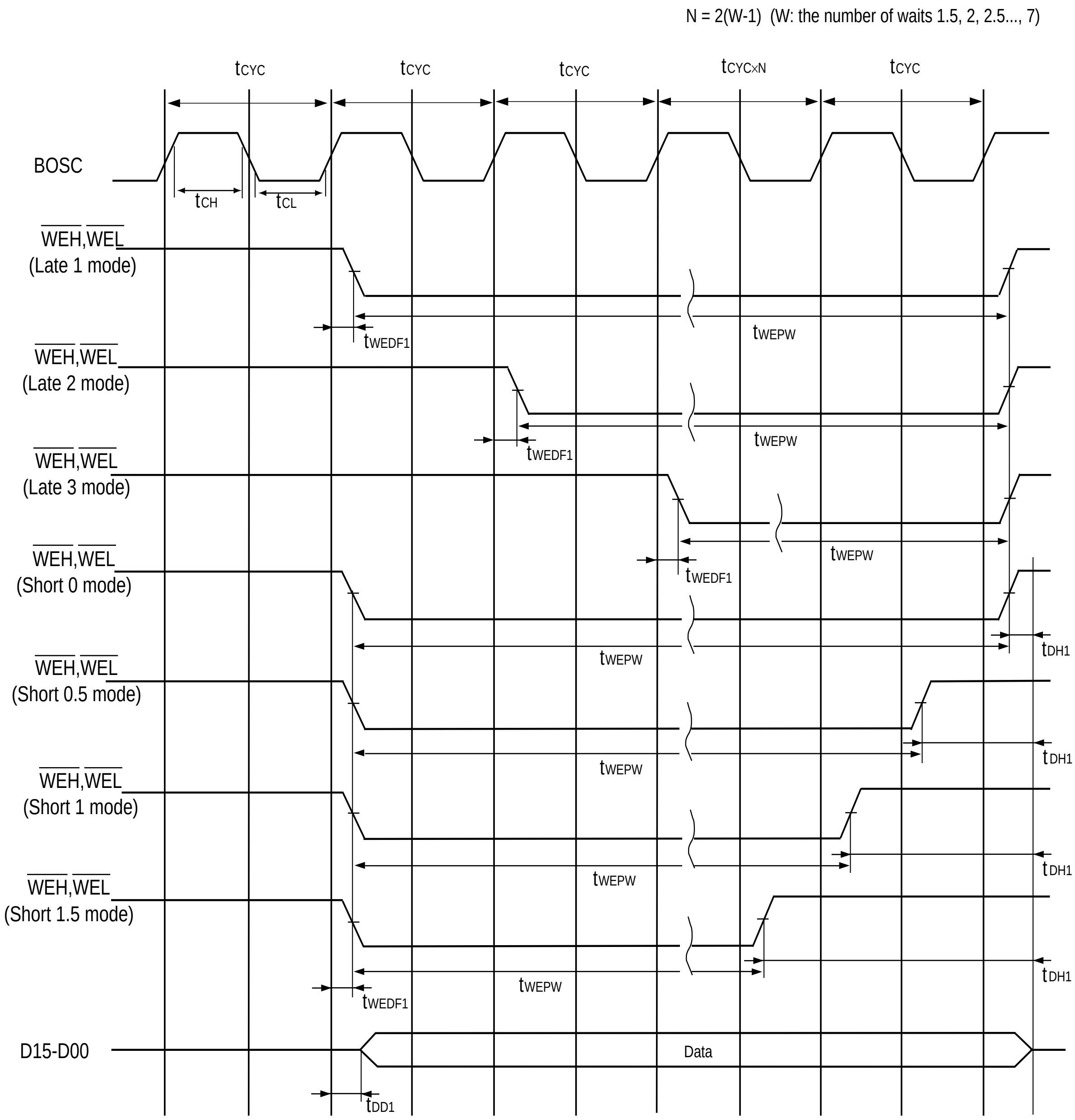


Figure 11-1-7 Data Transfer Signal Timing
(Address/Data Separate, With Wait (1.5 or More), /WE Late, Short Mode)

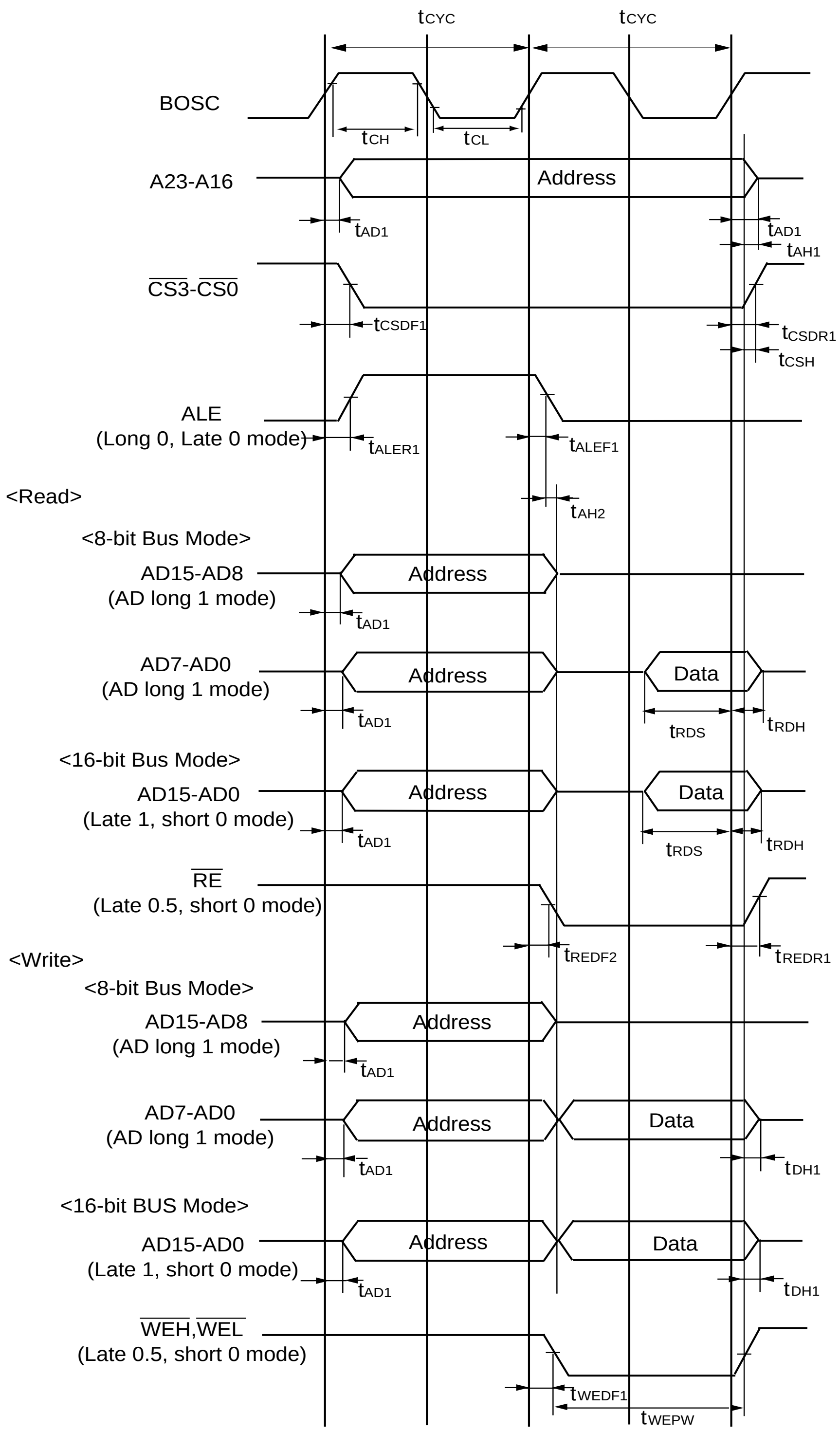


Figure 11-1-8 Data Transfer Signal Timing
(Address/Data Shared, Without Wait, Read/Write)

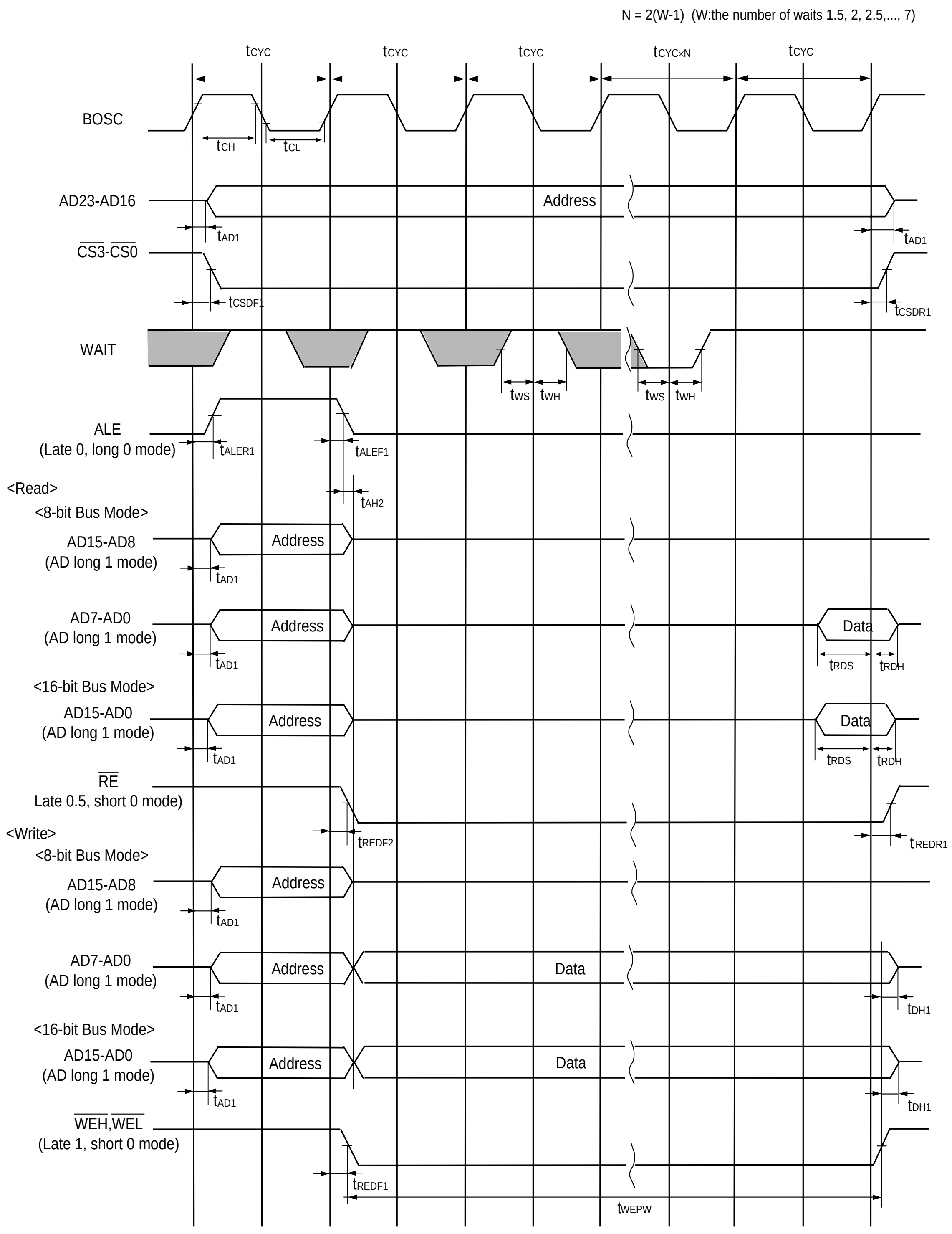


Figure 11-1-9 Data Transfer Signal Timing
(Address/Data Shared, With Wait (1.5 or More), Read/Write)

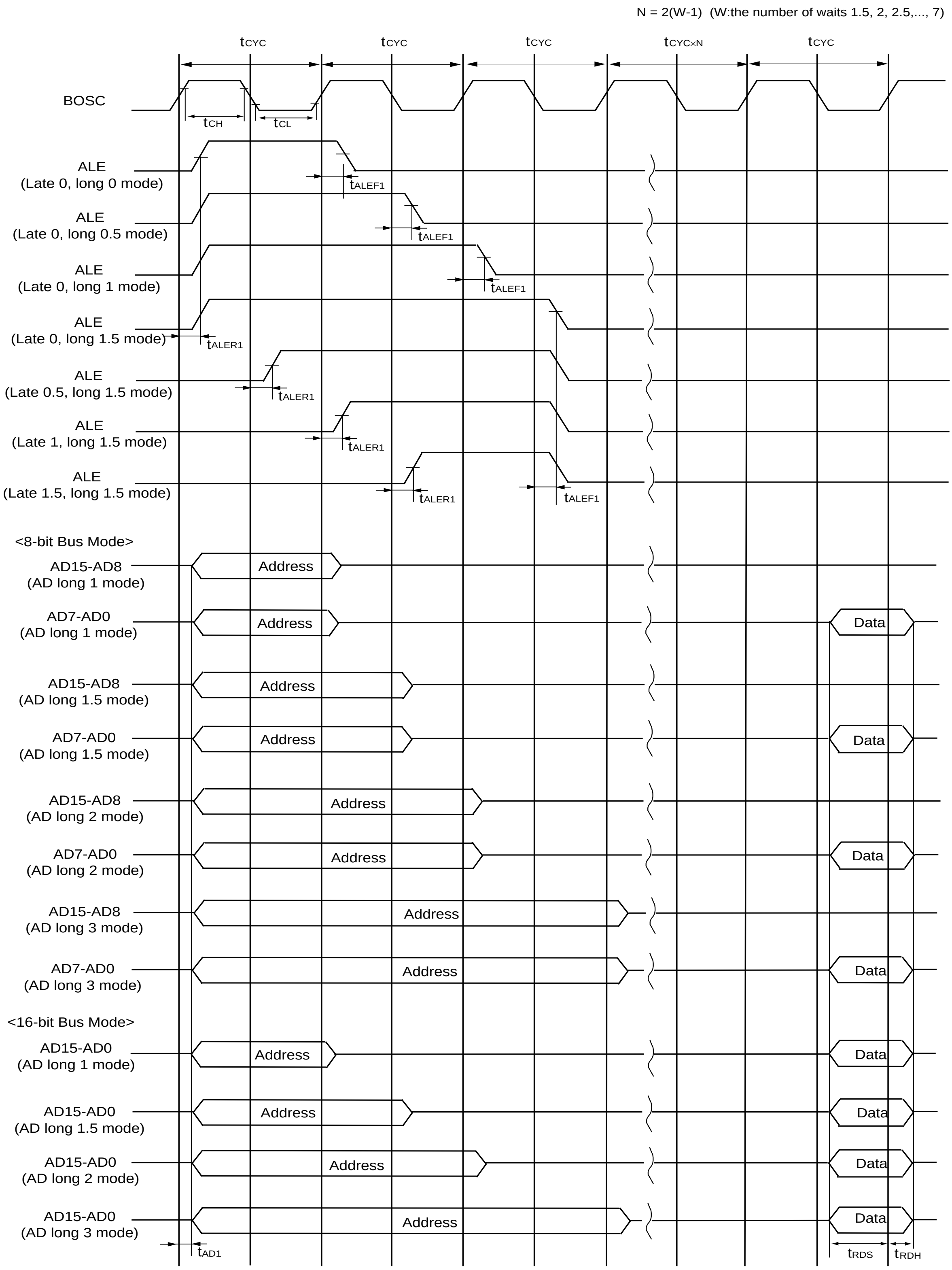


Figure 11-1-10 Data Transfer Signal Timing
(Address/Data Shared, With Wait (1.5 or More), ALE late, long mode, /AD long mode, Read)

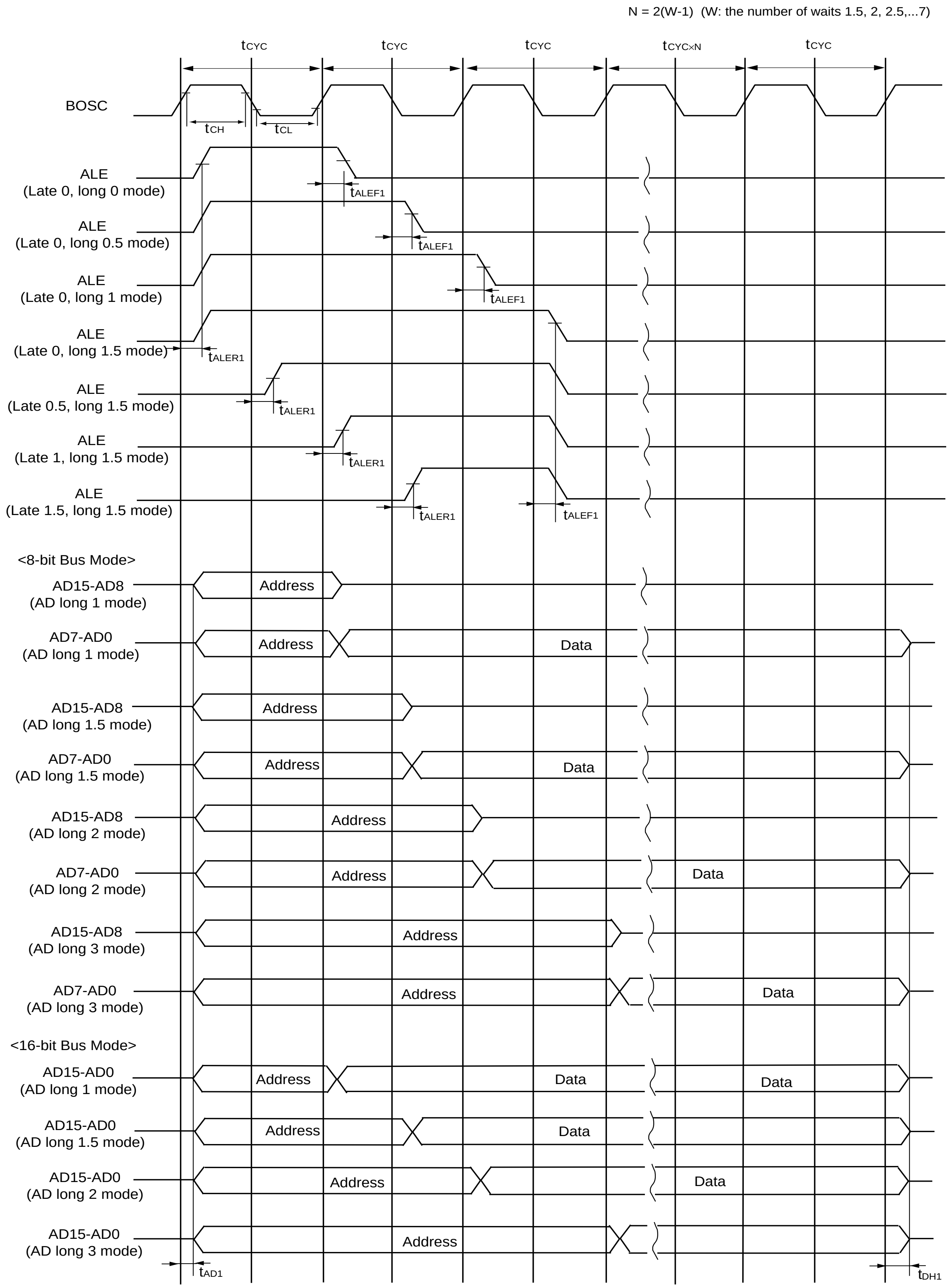


Figure 11-1-11 Data Transfer Signal Timing
(Address/Data Shared, With Wait (1.5 or More), ALE late, long mode, /AD long mode, Write)

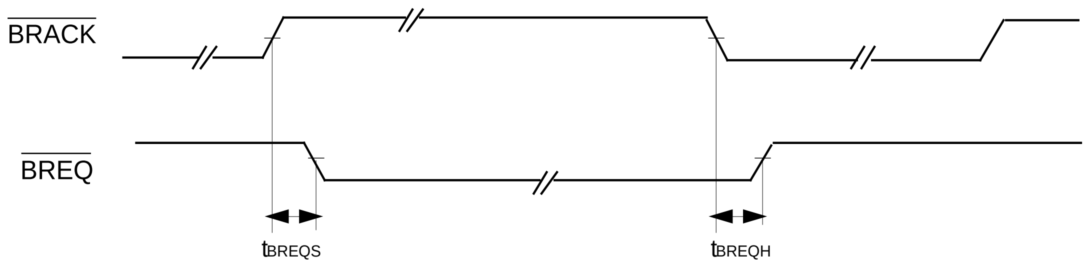


Figure 11-1-12 Bus Authority Request Signal Timing

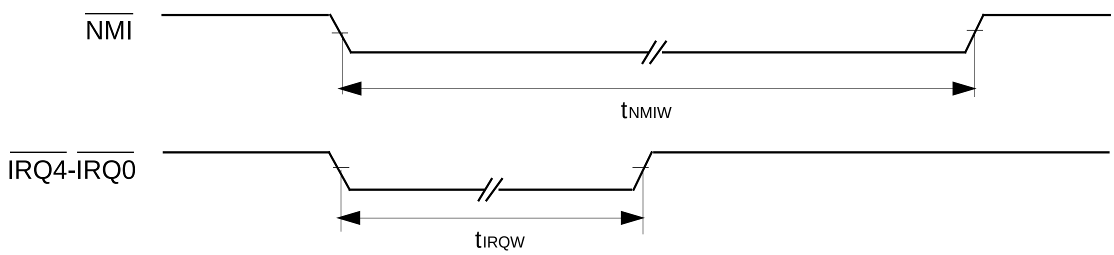


Figure 11-1-13 Interrupt Signal Timing

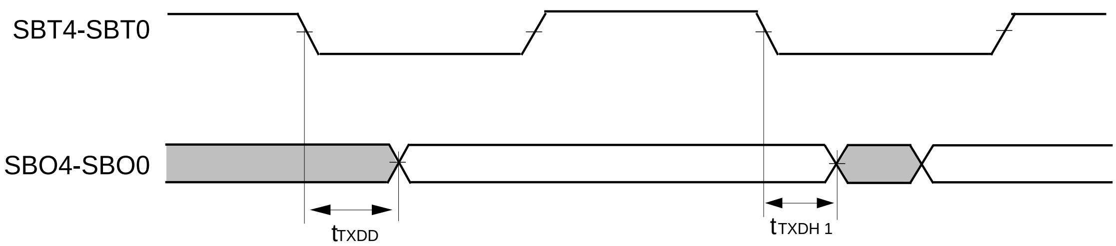


Figure 11-1-14 Serial Interface Signal Timing 1
(Synchronous Serial Transmission: Transfer in Progress)

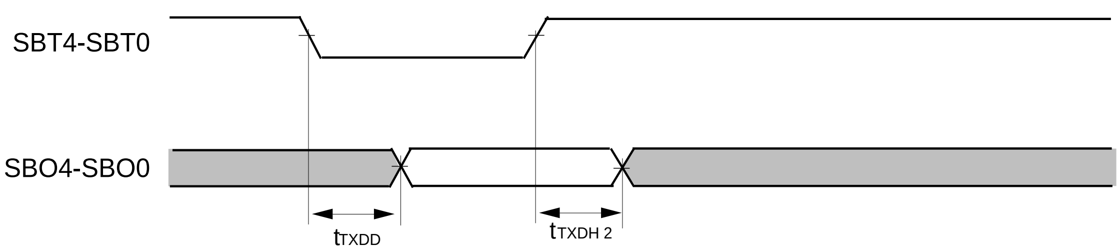


Figure 11-1-15 Serial Interface Signal Timing 2
(Synchronous Serial Transmission: Transfer End Timing at SBT Input)

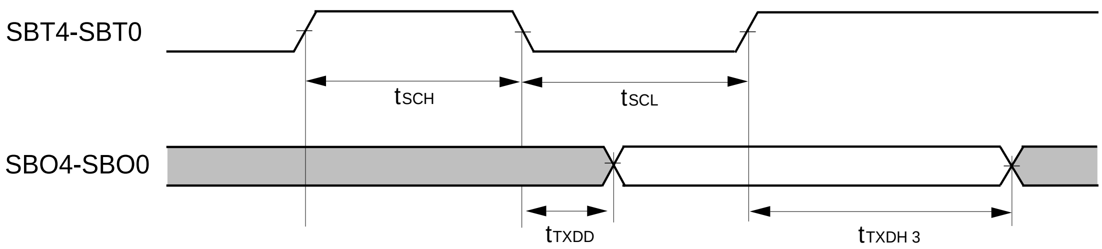


Figure 11-1-16 Serial Interface Signal Timing 3
(Synchronous Serial Transmission: Transfer End Timing at SBT Output)

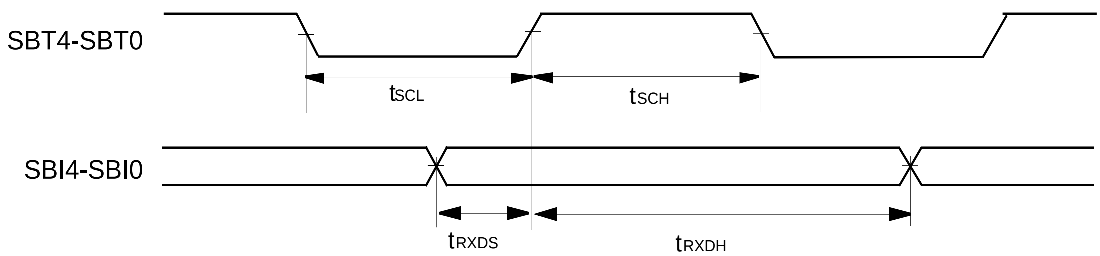


Figure 11-1-17 Serial Interface Signal Timing 4
(Synchronous Serial Reception: Transfer End Timing at SBT Input)

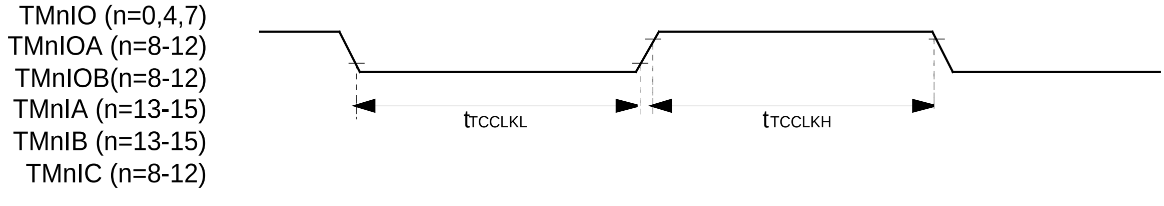


Figure 11-1-18 Timer/Counter Signal Timing

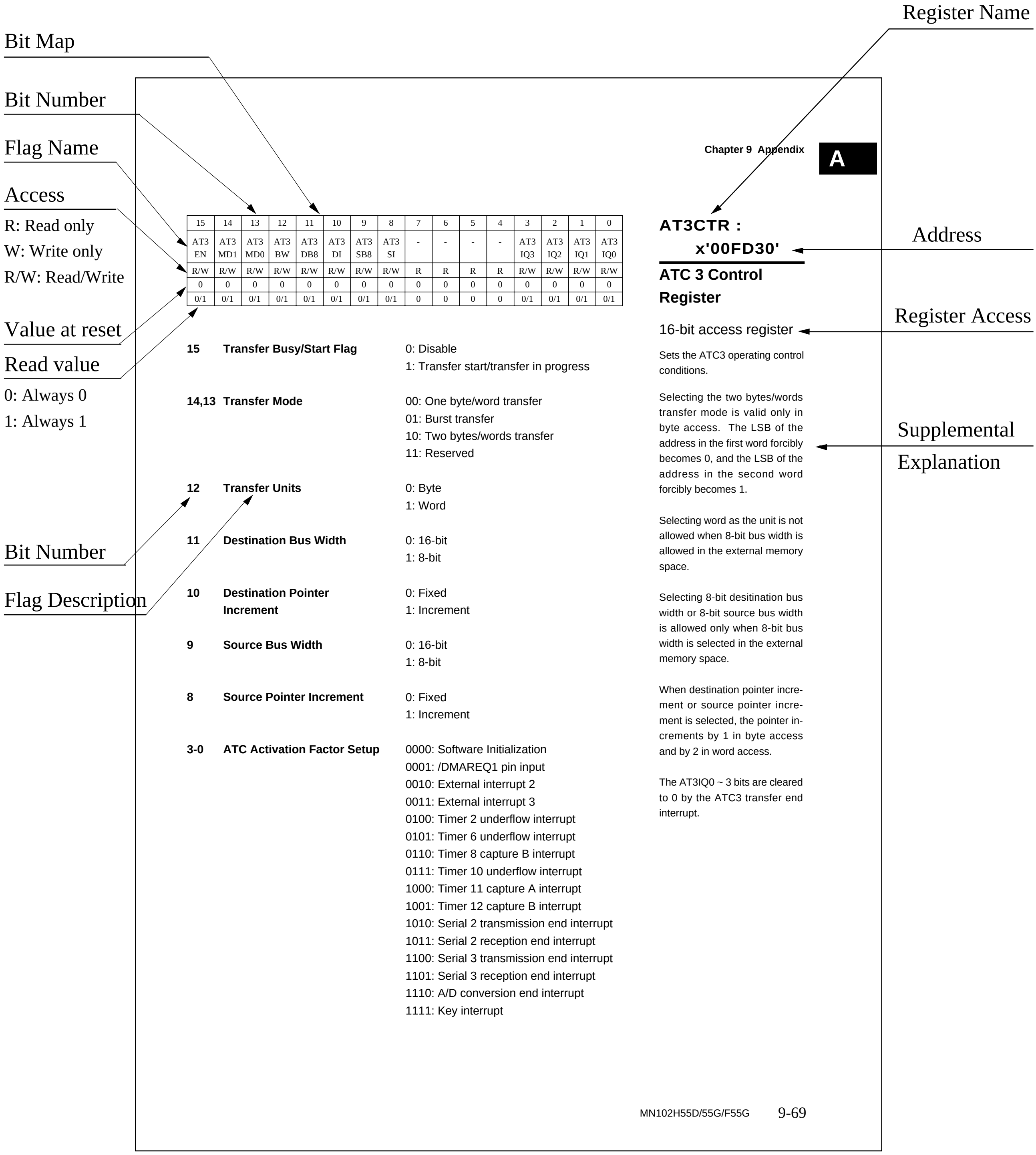
11-2 Data Appendix

11-2-1 List of Special Registers

About This Section

■ Description of Each Page

Each page of this chapter describes one or more registers. Each page lists the register name, address, register access, bit map, flag explanation of each bit number and supplementary explanation. The following is the layout and definition of this section.



Chapter 9 Appendix **A**

AT3CTR : x'00FD30'

ATC 3 Control Register

16-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AT3 EN	AT3 MD1	AT3 MD0	AT3 BW	AT3 DB8	AT3 DI	AT3 SB8	AT3 SI	-	-	-	-	AT3 IQ3	AT3 IQ2	AT3 IQ1	AT3 IQ0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0	0	0	0	0/1	0/1	0/1	0/1

15	**Transfer Busy/Start Flag**	0: Disable 1: Transfer start/transfer in progress
14,13	**Transfer Mode**	00: One byte/word transfer 01: Burst transfer 10: Two bytes/words transfer 11: Reserved
12	**Transfer Units**	0: Byte 1: Word
11	**Destination Bus Width**	0: 16-bit 1: 8-bit
10	**Destination Pointer Increment**	0: Fixed 1: Increment
9	**Source Bus Width**	0: 16-bit 1: 8-bit
8	**Source Pointer Increment**	0: Fixed 1: Increment
3-0	**ATC Activation Factor Setup**	0000: Software Initialization 0001: /DMAREQ1 pin input 0010: External interrupt 2 0011: External interrupt 3 0100: Timer 2 underflow interrupt 0101: Timer 6 underflow interrupt 0110: Timer 8 capture B interrupt 0111: Timer 10 underflow interrupt 1000: Timer 11 capture A interrupt 1001: Timer 12 capture B interrupt 1010: Serial 2 transmission end interrupt 1011: Serial 2 reception end interrupt 1100: Serial 3 transmission end interrupt 1101: Serial 3 reception end interrupt 1110: A/D conversion end interrupt 1111: Key interrupt

Sets the ATC3 operating control conditions.

Selecting the two bytes/words transfer mode is valid only in byte access. The LSB of the address in the first word forcibly becomes 0, and the LSB of the address in the second word forcibly becomes 1.

Selecting word as the unit is not allowed when 8-bit bus width is allowed in the external memory space.

Selecting 8-bit desitination bus width or 8-bit source bus width is allowed only when 8-bit bus width is selected in the external memory space.

When destination pointer increment or source pointer increment is selected, the pointer increments by 1 in byte access and by 2 in word access.

The AT3IQ0 ~ 3 bits are cleared to 0 by the ATC3 transfer end interrupt.

MN102H55D/55G/F55G 9-69

C

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WD RST	WD M1	WD M0	-	-	-	-	-	-	-	-	OSC ID	STOP	HALT	OSC1	OSC0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1

CPUM : x'00FC00'

CPU Mode Control Register

16-bit access register

CPUM controls the CPU modes and oscillator for watchdog timer.

15 Watchdog Timer Enable 0: Enable 1: Disable and clear

14:13 Watchdog Timer Count *

00: 2^{17}
01: 2^{5}
10: 2^{13}
11: 2^{15}

Shorten the oscillation wait time from STOP mode and a watchdog timer interrupt cycle. The same counter is used for setting both values. Set the WDREG register to extend the watchdog timer interrupt cycle additionally.

bit 13, 14	Watchdog Interrupt Cycle (BOSC Signal)	Return Time From STOP Mode
00	2^{17} cycles	$2^{17}\times$(1/fosci)
01	2^{5} cycles	$2^{5}\times$(1/fosci)
10	2^{13} cycles	$2^{13}\times$(1/fosci)
11	2^{15} cycles	$2^{15}\times$(1/fosci)

* In the MN102HF55G, set these bits to only '00'.

4 System Clock Monitor

0: OSCI input
1: Low-speed clock input

3:2 CPU Operating Control

1:0 Oscillator Control

STOP	HALT	OSC1	OSC0	CPU Mode	CPU	Clock	OSCID Value
0	0	0	0	NORMAL	On	OSCI	0
0	0	1	1	SLOW	On	XI	1
0	1	0	0	HALT0	On	OSCI	0
0	1	1	1	HALT1	On	XI	1
1	0	0	0	STOP0	Off	-	-
1	0	1	1	STOP1	Off	-	-

The following describes programming rules and precautions in the STOP/HALT mode.

Points for Programming

(1) Setting the CPUM address in the address register in advance, set the CPUM register using the MOV instruction with the register indirect addressing mode.
(2) Immediately after the MOV instruction, locate three NOPs consecutively.
(3) Immediately before the MOV instruction, locate the JMP instruction and align to the even address. This avoids the effects by the differences of the bus widths in the memory mode or expansion mode and provides the same result when operating in any conditions.

Programming Coding Example in Assembler (as 102Ver.1.0, Ver.2.0)

```
          MOV    CPUM, A0    ; Set A0 to the CPUM address.
          MOV    (A0), D0    ; Transfer the contents of CPUM to D0.
          OR     x'000*', D0 ; Generate the data to set the STOP/HALT mode.
          JMP    STP_HLT     ; Branch unconditionally to the even address  to
          ALIGN  2           ; eliminate the difference of operating conditions.
STP_HLT   MOV    D0, (A0)    ; Set the STOP/HALT mode to CPUM.
          NOP                ; Dummy
          NOP                ; Dummy
          NOP                ; Dummy
```

Precautions

(1) * of OR instruction varies depending on the STOP or HALT mode.
(2) Set the ALIGN value to '2' or more in the above file when the ALIGN value is set using SECTION dummy instruction before this programming coding is described.
(3) Code the above programming in another file of the assembler source file when the program is developed with C complier cc 102.

EFCR : x'00FC08'

Expansion Control Register

16-bit access register

EFCR sets 2-byte/1-cycle decode mode.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	STEN	QDEC
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0/1	0/1

1 **Saturation Operation Setup**
0: Disable (normal operation)
1: Enable (saturation operation when the ST flag of the PSW register is 1.)

0 **2 bytes/1 cycle Decode Setup**
0: Disable (Decode at the same cycles in the MN102L00 series.
1: Enable (Decode the 2-byte instruction at high speed.)

IAGR : x'00FC0E'

Interrupt Accept Group Register

16-bit access register

IAGR returns the group number of the accepted interrupt.

IAGR stores the group number of the accepted interrupt. The 6-bit GN field indicates the group number. When the first address of the interrupt service routine, add the contents of the IAGR register to the first address of the table in which registered vector address for each interrupt servicing. The IAGR register is only read.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	GN5	GN4	GN3	GN2	GN1	GN0	-	-
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0	0

7:2 **Returns the group number multiplied by 4**

NMICR : x'00FC40'

Nonmaskable Interrupt Control Register

8-bit access register

NMICR verifies a nonmaskable interrupt.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	NMID
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Nonmaskable Interrupt Detect Flag**	0: No interrupt detected 1: Interrupt detected

E

I

WDICR : x'00FC42'

Watchdog Interrupt Control Register

8-bit access register

WDICR verifies a watchdog interrupt.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	WDID
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Watchdog Interrupt Detect Flag**	0: No interrupt detected 1: Interrupt detected

N

UNICR : x'00FC44'

Undefined Instruction Interrupt Control Register

8-bit access register

UNICR verifies an undefined instruction interrupt.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	UNID
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Undefined Instruction Interrupt Detect Flag**	0: No interrupt detected 1: Interrupt detected

U

W

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-
R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

EIICR : x'00FC46'

Error Interrupt Control Register

8-bit access register

EIICR verifies an error interrupt.

This register does not exist. When an interrupt vector is not determined, this register indicates an error by writing IAGR register to 'C'.

7	6	5	4	3	2	1	0
-	-	-	IQ0 IR	-	-	-	IQ0 ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

IQ0ICL : x'00FC50'

External Interrupt 0 Control Register

8-bit access register

IQ0ICL requests and verifies an external interrupt 0 interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data.

E

I

4	**External Interrupt 0 Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**External Interrupt 0 Detect Flag**	0: No interrupt detected 1: Interrupt detected

7	6	5	4	3	2	1	0
-	IQ0 LV2	IQ0 LV1	IQ0 LV0	-	-	-	IQ0 IE
R	R/W	R/W	R/W	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0/1	0/1	0/1	0	0	0	0/1

IQ0ICH : x'00FC51'

External Interrupt 0 Control Register

8-bit access register

IQ0ICH sets an external interrupt 0 interrupt level and enables an interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data.

6:4	**External Interrupt 0 Level Setup**	Set the level from 0 to 6
0	**External Interrupt 0 Interrupt Enable Flag**	0: Disable 1: Enable

TM0UICL : x'00FC52'

Timer 0 Underflow Interrupt Control Register

8-bit access register

TM0UICL requests and verifies a timer 0 interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM0UIR	-	-	-	TM0UID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Timer 0 Underflow Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 0 Underflow Interrupt Detect Flag**	0: No interrupt detected 1: Interrupt detected

TM0UICH : x'00FC53'

Timer 0 Underflow Interrupt Control Register

8-bit access register

TM0UICH enables a timer 0 interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data. The interrupt level is the same level set in the IQ0LV[2:0] bits of the IQ0ICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM0UIE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Timer 0 Underflow Interrupt Enable Flag**	0: Disable 1: Enable

TM8UICL : x'00FC54'

Timer 8 Underflow Interrupt Control Register

8-bit access register

TM8UICL requests and verifies a timer 8 interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM8U IR	-	-	-	TM8U ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

Bit	Name	Setting
4	**Timer 8 Underflow Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 8 Underflow Interrupt Detect Flag**	0: No interrupt detected 1: Interrupt detected

TM8UICH : x'00FC55'

Timer 8 Underflow Interrupt Control Register

8-bit access register

TM8UICH enables a timer 8 interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data. The interrupt level is the same level set in the IQ0LV[2:0] bits of the IQ0ICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM8U IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

Bit	Name	Setting
0	**Timer 8 Underflow Interrupt Enable Flag**	0: Disable 1: Enable

TM8AICL : x'00FC56'

Timer 8 Capture A Interrupt Control Register

8-bit access register

TM8AICL requests and verifies a timer 8 capture A interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM8A IR	-	-	-	TM8A ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

Bit	Name	Description
4	**Timer 8 Capture A Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 8 Capture A Interrupt Detect Flag**	0: No interrupt detected 1: Interrupt detected

TM8AICH : x'00FC57'

Timer 8 Capture A Interrupt Control Register

8-bit access register

TM8AICH enables a timer 8 capture A interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data. The interrupt level is the same level set in the IQ0LV[2:0] bits of the IQ0ICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM8A IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

Bit	Name	Description
0	**Timer 8 Capture A Interrupt Enable Flag**	0: Disable 1: Enable

IQ1ICL : x'00FC58'

External Interrupt 1 Control Register

8-bit access register

IQ1ICL requests and verifies an external interrupt 1 interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	IQ1 IR	-	-	-	IQ1 ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4 **External Interrupt 1 Request Flag**
0: No interrupt requested
1: Interrupt requested

0 **External Interrupt 1 Detect Flag**
0: No interrupt detected
1: Interrupt detected

I

IQ1ICH : x'00FC59'

External Interrupt 1 Control Register

8-bit access register

IQ1ICH sets an external interrupt 1 interrupt level and enables an interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	IQ1 LV2	IQ1 LV1	IQ1 LV0	-	-	-	IQ1 IE
R	R/W	R/W	R/W	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0/1	0/1	0/1	0	0	0	0/1

6:4 **External Interrupt 1 Level Setup**
Set the level from 0 to 6

0 **External Interrupt 1 Interrupt Enable Flag**
0: Disable
1: Enable

T

TM1UICL : x'00FC5A'

Timer 1 Underflow Interrupt Control Register

8-bit access register

TM1UICL requests and verifies a timer 1 interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM1U IR	-	-	-	TM1U ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Timer 1 Underflow Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 1 Underflow Interrupt Detect Flag**	0: No interrupt detected 1: Interrupt detected

TM1UICH : x'00FC5B'

Timer 1 Underflow Interrupt Control Register

8-bit access register

TM1UICH enables a timer 1 interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data. The interrupt level is the same level set in the IQ1LV[2:0] bits of the IQ1ICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM1U IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Timer 1 Underflow Interrupt Enable Flag**	0: Disable 1: Enable

TM8BICL : x'00FC5C'

Timer 8 Capture B Interrupt Control Register

8-bit access register

TM8BICL requests and verifies a timer 8 capture B interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM8B IR	-	-	-	TM8B ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Timer 8 Capture B Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 8 Capture B Interrupt Detect Flag**	0: No interrupt detected 1: Interrupt detected

TM8BICH : x'00FC5D'

Timer 8 Capture B Interrupt Control Register

8-bit access register

TM8BICH enables a timer 8 capture B interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data. The interrupt level is the same level set in the IQ1LV[2:0] bits of the IQ1ICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM8B IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Timer 8 Capture B Interrupt Enable Flag**	0: Disable 1: Enable

T

TM9UICL : x'00FC5E'

Timer 9 Underflow Interrupt Control Register

8-bit access register

TM9UICL requests and verifies a timer 9 interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM9UIR	-	-	-	TM9UID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Timer 9 Underflow Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 9 Underflow Interrupt Detect Flag**	0: No interrupt detected 1: Interrupt detected

TM9UICH : x'00FC5F'

Timer 9 Underflow Interrupt Control Register

8-bit access register

TM9UICH enables a timer 9 interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data. The interrupt level is the same level set in the IQ1LV[2:0] bits of the IQ1ICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM9UIE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Timer 9 Underflow Interrupt Enable Flag**	0: Disable 1: Enable

IQ2ICL : x'00FC60'

External Interrupt 2 Control Register

8-bit access register

IQ2ICL requests and verifies an external interrupt 2 interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	IQ2 IR	-	-	-	IQ2 ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**External Interrupt 2 Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**External Interrupt 2 Detect Flag**	0: No interrupt detected 1: Interrupt detected

I

IQ2ICH : x'00FC61'

External Interrupt 2 Control Register

8-bit access register

IQ2ICH sets an external interrupt 2 interrupt level and enables an interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	IQ2 LV2	IQ2 LV1	IQ2 LV0	-	-	-	IQ2 IE
R	R/W	R/W	R/W	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0/1	0/1	0/1	0	0	0	0/1

6:4	**External Interrupt 2 Level Setup**	Set the level from 0 to 6
0	**External Interrupt 2 Interrupt Enable Flag**	0: Disable 1: Enable

T

7	6	5	4	3	2	1	0
-	-	-	TM2U IR	-	-	-	TM2U ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

TM2UICL : x'00FC62'

Timer 2 Underflow Interrupt Control Register

8-bit access register

TM2UICL requests and verifies a timer 2 interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data.

4	**Timer 2 Underflow Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 2 Underflow Interrupt Detect Flag**	0: No interrupt detected 1: Interrupt detected

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM2U IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

TM2UICH : x'00FC63'

Timer 2 Underflow Interrupt Control Register

8-bit access register

TM2UICH enables a timer 2 interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data. The interrupt level is the same level set in the IQ2LV[2:0] bits of the IQ2ICH register.

0	**Timer 2 Underflow Interrupt Enable Flag**	0: Disable 1: Enable

TM9AICL : x'00FC64'

Timer 9 Capture A Interrupt Control Register

8-bit access register

TM9AICL requests and verifies a timer 9 capture A interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM9A IR	-	-	-	TM9A ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4 Timer 9 Capture A Interrupt Request Flag
0: No interrupt requested
1: Interrupt requested

0 Timer 9 Capture A Interrupt Detect Flag
0: No interrupt detected
1: Interrupt detected

TM9AICH : x'00FC65'

Timer 9 Capture A Interrupt Control Register

8-bit access register

TM9AICH enables a timer 9 capture A interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data. The interrupt level is the same level set in the IQ2LV[2:0] bits of the IQ2ICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM9A IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0 Timer 9 Capture A Interrupt Enable Flag
0: Disable
1: Enable

TM9BICL : x'00FC66'

Timer 9 Capture B Interrupt Control Register

8-bit access register

TM9BICL requests and verifies a timer 9 capture B interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM9B IR	-	-	-	TM9B ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Timer 9 Capture B Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 9 Capture B Interrupt Detect Flag**	0: No interrupt detected 1: Interrupt detected

TM9BICH : x'00FC67'

Timer 9 Capture B Interrupt Control Register

8-bit access register

TM9BICH enables a timer 9 capture B interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data. The interrupt level is the same level set in the IQ2LV[2:0] bits of the IQ2ICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM9B IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Timer 9 Capture B Interrupt Enable Flag**	0: Disable 1: Enable

IQ3ICL : x'00FC68'

External Interrupt 3 Control Register

8-bit access register

IQ3ICL requests and verifies an external interrupt 3 interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	IQ3 IR	-	-	-	IQ3 ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**External Interrupt 3 Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**External Interrupt 3 Detect Flag**	0: No interrupt detected 1: Interrupt detected

I

IQ3ICH : x'00FC69'

External Interrupt 3 Control Register

8-bit access register

IQ3ICH sets an external interrupt 3 interrupt level and enables an interrupt.

This register allows only byte-accesses. Use MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	IQ3 LV2	IQ3 LV1	IQ3 LV0	-	-	-	IQ3 IE
R	R/W	R/W	R/W	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0/1	0/1	0/1	0	0	0	0/1

6:4	**External Interrupt 3 Level Setup**	Set the level from 0 to 6
0	**External Interrupt 3 Interrupt Enable Flag**	0: Disable 1: Enable

T

TM3UICL : x'00FC6A'

Timer 3 Underflow Interrupt Control Register

8-bit access register

TM3UICL requests and verifies a timer 3 interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM3U IR	-	-	-	TM3U ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Timer 3 Underflow Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 3 Underflow Interrupt Detect Flag**	0: No interrupt detected 1: Interrupt detected

TM3UICH : x'00FC6B'

Timer 3 Underflow Interrupt Control Register

8-bit access register

TM3UICH enables a timer 3 interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the IQ3LV[2:0] bits of the IQ3ICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM3U IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Timer 3 Underflow Interrupt Enable Flag**	0: Disable 1: Enable

TM10UICL : x'00FC6C'

Timer 10 Underflow Interrupt Control Register

8-bit access register

TM10UICL requests and verifies a timer 10 interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM10UIR	-	-	-	TM10UID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Timer 10 Underflow Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 10 Underflow Interrupt Detect Flag**	0: No interrupt detected 1: Interrupt detected

TM10UICH : x'00FC6D'

Timer 10 Underflow Interrupt Control Register

8-bit access register

TM10UICH enables a timer 10 interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the IQ3LV[2:0] bits of the IQ3ICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM10UIE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Timer 10 Underflow Interrupt Enable Flag**	0: Disable 1: Enable

T

TM10AICL : x'00FC6E'

Timer 10 Capture A Interrupt Control Register

8-bit access register

TM10AICL requests and verifies a timer 10 capture A interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM10A IR	-	-	-	TM10A ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Timer 10 Capture A Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 10 Capture A Interrupt Detect Flag**	0: No interrupt detected 1: Interrupt detected

TM10AICH : x'00FC6F'

Timer 10 Capture A Interrupt Control Register

8-bit access register

TM10AICH enables a timer 10 capture A interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the IQ3LV[2:0] bits of the IQ3ICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM10A IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Timer 10 Capture A Interrupt Enable Flag**	0: Disable 1: Enable

7	6	5	4	3	2	1	0
-	-	-	IQ4 IR	-	-	-	IQ4 ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

IQ4ICL : x'00FC70'

External Interrupt 4 Control Register

8-bit access register

IQ4ICL requests and verifies an external interrupt 4 interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

4 **External Interrupt 4 Request Flag** — 0: No interrupt requested; 1: Interrupt requested

0 **External Interrupt 4 Detect Flag** — 0: No interrupt detected; 1: Interrupt detected

I

7	6	5	4	3	2	1	0
-	IQ4 LV2	IQ4 LV1	IQ4 LV0	-	-	-	IQ4 IE
R	R/W	R/W	R/W	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0/1	0/1	0/1	0	0	0	0/1

IQ4ICH : x'00FC71'

External Interrupt 4 Control Register

8-bit access register

IQ4ICH sets an external interrupt 4 interrupt level and enables an interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

6:4 **External Interrupt 4 Level Setup** — Set the level from 0 to 6

0 **External Interrupt 4 Interrupt Enable Flag** — 0: Disable; 1: Enable

T

TM4UICL : x'00FC72'

Timer 4 Underflow Interrupt Control Register

8-bit access register

TM4UICL requests and verifies a timer 4 interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM4U IR	-	-	-	TM4U ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

Bit	Name	Setting
4	**Timer 4 Underflow Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 4 Underflow Interrupt Detect Flag**	0: No interrupt detected 1: Interrupt detected

TM4UICH : x'00FC73'

Timer 4 Underflow Interrupt Control Register

8-bit access register

TM4UICH enables a timer 4 interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the IQ4LV[2:0] bits of the IQ4ICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM4U IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

Bit	Name	Setting
0	**Timer 4 Underflow Interrupt Enable Flag**	0: Disable 1: Enable

TM10BICL : x'00FC74'

Timer 10 Capture B Interrupt Control Register

8-bit access register

TM10BICL requests and verifies a timer 10 capture B interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM10B IR	-	-	-	TM10B ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Timer 10 Capture B Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 10 Capture B Interrupt Detect Flag**	0: No interrupt detected 1: Interrupt detected

TM10BICH : x'00FC75'

Timer 10 Capture B Interrupt Control Register

8-bit access register

TM10BICH enables a timer 10 capture B interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the IQ4LV[2:0] bits of the IQ4ICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM10B IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Timer 10 Capture B Interrupt Enable Flag**	0: Disable 1: Enable

T

TM11UICL : x'00FC76'

Timer 11 Underflow Interrupt Control Register

8-bit access register

TM11UICL requests and verifies a timer 11 interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM11UIR	-	-	-	TM11UID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4 **Timer 11 Underflow Interrupt Request Flag**
0: No interrupt requested
1: Interrupt requested

0 **Timer 11 Underflow Interrupt Detect Flag**
0: No interrupt detected
1: Interrupt detected

TM11UICH : x'00FC77'

Timer 11 Underflow Interrupt Control Register

8-bit access register

TM11UICH enables a timer 11 interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the IQ4LV[2:0] bits of the IQ4ICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM11UIE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0 **Timer 11 Underflow Interrupt Enable Flag**
0: Disable
1: Enable

KIICL : x'00FC78'

External Key Interrupt Control Register

8-bit access register

KIICL requests and verifies an external key interrupt interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	KI IR	-	-	-	KI ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**External Key Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**External Key Interrupt Detect Flag**	0: No interrupt detected 1: Interrupt detected

K

KIICH : x'00FC79'

External Key Interrupt Control Register

8-bit access register

KIICH sets an external key interrupt level and enables an interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	KI LV2	KI LV1	KI LV0	-	-	-	KI IE
R	R/W	R/W	R/W	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0/1	0/1	0/1	0	0	0	0/1

6:4	**External Key Interrupt Level Setup**	Set the level from 0 to 6
0	**External Key Interrupt Enable Flag**	0: Disable 1: Enable

T

7	6	5	4	3	2	1	0
-	-	-	TM5U IR	-	-	-	TM5U ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

TM5UICL : x'00FC7A'

Timer 5 Underflow Interrupt Control Register

8-bit access register

TM5UICL requests and verifies a timer 5 interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

4	**Timer 5 Underflow Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 5 Underflow Interrupt Detect Flag**	0: No interrupt detected 1: Interrupt detected

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM5U IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

TM5UICH : x'00FC7B'

Timer 5 Underflow Interrupt Control Register

8-bit access register

TM5UICH enables a timer 5 interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the KILV[2:0] bits of the KIICH register.

0	**Timer 5 Underflow Interrupt Enable Flag**	0: Disable 1: Enable

TM11AICL : x'00FC7C'

Timer 11 Capture A Interrupt Control Register

8-bit access register

TM11AICL requests and verifies a timer 11 capture A interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM11A IR	-	-	-	TM11A ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Timer 11 Capture A Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 11 Capture A Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

TM11AICH : x'00FC7D'

Timer 11 Capture A Interrupt Control Register

8-bit access register

TM11AICH enables a timer 11 capture A interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the KILV[2:0] bits of the KIICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM11A IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Timer 11 Capture A Interrupt Enable Flag**	0: Disable 1: Enable

TM11BICL : x'00FC7E'

Timer 11 Capture B Interrupt Control Register

8-bit access register

TM11BICL requests and verifies a timer 11 capture B interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM11B IR	-	-	-	TM11B ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Timer 11 Capture B Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 11 Capture B Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

TM11BICH : x'00FC7F'

Timer 11 Capture B Interrupt Control Register

8-bit access register

TM11BICH enables a timer 11 capture B interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the KILV[2:0] bits of the KIICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM11B IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Timer 11 Capture B Interrupt Enable Flag**	0: Disable 1: Enable

7	6	5	4	3	2	1	0
-	-	-	AD IR	-	-	-	AD ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

ADICL : x'00FC80'

AD Conversion End Interrupt Control Register

8-bit access register

ADICL requests and verifies an AD conversion end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

4	**AD Conversion End Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**AD Conversion End Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

7	6	5	4	3	2	1	0
-	AD LV2	AD LV1	AD LV0	-	-	-	AD IE
R	R/W	R/W	R/W	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0/1	0/1	0/1	0	0	0	0/1

ADICH : x'00FC81'

AD Conversion End Interrupt Control Register

8-bit access register

ADICH sets an AD conversion end interrupt level and enables an interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

6:4	**AD Conversion End Interrupt Level Setup**	Set the level from 0 to 6
0	**AD Conversion End Interrupt Enable Flag**	0: Disable 1: Enable

TM6UICL : x'00FC82'

Timer 6 Underflow Interrupt Control Register

8-bit access register

TM6UICL requests and verifies a timer 6 interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM6UIR	-	-	-	TM6UID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Timer 6 Underflow Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 6 Underflow Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

TM6UICH : x'00FC83'

Timer 6 Underflow Interrupt Control Register

8-bit access register

TM6UICH enables a timer 6 interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the ADLV[2:0] bits of the ADICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM6UIE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Timer 6 Underflow Interrupt Enable Flag**	0: Disable 1: Enable

TM12UICL : x'00FC84'

Timer 12 Underflow Interrupt Control Register

8-bit access register

TM12UICL requests and verifies a timer 12 interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM12U IR	-	-	-	TM12U ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Timer 12 Underflow Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 12 Underflow Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

TM12UICH : x'00FC85'

Timer 12 Underflow Interrupt Control Register

8-bit access register

TM12UICH enables a timer 12 interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the ADLV[2:0] bits of the ADICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM12U IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Timer 12 Underflow Interrupt Enable Flag**	0: Disable 1: Enable

TM12AICL : x'00FC86'

Timer 12 Capture A Interrupt Control Register

8-bit access register

TM12AICL requests and verifies a timer 12 capture A interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM12A IR	-	-	-	TM12A ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Timer 12 Capture A Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 12 Capture A Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

TM12AICH : x'00FC87'

Timer 12 Capture A Interrupt Control Register

8-bit access register

TM12AICH enables a timer 12 capture A interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the ADLV[2:0] bits of the ADICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM12A IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Timer 12 Capture A Interrupt Enable Flag**	0: Disable 1: Enable

TM7UICL : x'00FC88'

Timer 7 Underflow Interrupt Control Register

8-bit access register

TM7UICL requests and verifies a timer 7 interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM7U IR	-	-	-	TM7U ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Timer 7 Underflow Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 7 Underflow Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

TM7UICH : x'00FC89'

Timer 7 Underflow Interrupt Control Register

8-bit access register

TM7UICH enables a timer 7 interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	TM7U LV2	TM7U LV1	TM7U LV0	-	-	-	TM7U IE
R	R/W	R/W	R/W	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0/1	0/1	0/1	0	0	0	0/1

6:4	**Timer 7 Underflow Interrupt Level Setup**	Set the level from 0 to 6
0	**Timer 7 Underflow Interrupt Enable Flag**	0: Disable 1: Enable

TM12BICL : x'00FC8A'

Timer 12 Capture B Interrupt Control Register

8-bit access register

TM12BICL requests and verifies a timer 12 capture B interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	TM12B IR	-	-	-	TM12B ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Timer 12 Capture B Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Timer 12 Capture B Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

TM12BICH : x'00FC8B'

Timer 12 Capture B Interrupt Control Register

8-bit access register

TM12BICH enables a timer 12 capture B interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the TM7ULV[2:0] bits of the TM7UICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	TM12B IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Timer 12 Capture B Interrupt Enable Flag**	0: Disable 1: Enable

SC0TICL : x'00FC90'

Serial 0 Transmission End Interrupt Control Register

8-bit access register

SC0TICL requests and verifies a serial 0 transmission end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	SC0T IR	-	-	-	SC0T ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4 Serial 0 Transmission End Interrupt Request Flag
0: No interrupt requested
1: Interrupt requested

0 Serial 0 Transmission End Interrupt Detect Flag
0: Interrupt undetected
1: Interrupt detected

SC0TICH : x'00FC91'

Serial 0 Transmission End Interrupt Control Register

8-bit access register

SC0TICH sets a seial 0 transmission end interrupt level and enables an interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	SC0T LV2	SC0T LV1	SC0T LV0	-	-	-	SC0T IE
R	R/W	R/W	R/W	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0/1	0/1	0/1	0	0	0	0/1

6:4 Serial 0 Transmission End Interrupt Level Setup
Set the level from 0 to 6

0 Serial 0 Transmission End Interrupt Enable Flag
0: Disable
1: Enable

SC0RICL : x'00FC92'

Serial 0 Reception End Interrupt Control Register

8-bit access register

SC0RICL requests and verifies a seial 0 reception end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	SC0R IR	-	-	-	SC0R ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Serial 0 Reception End Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Serial 0 Reception End Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

SC0RICH : x'00FC93'

Serial 0 Reception End Interrupt Control Register

8-bit access register

SC0RICH enables a serial 0 reception end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the SC0TLV[2:0] bits of the SC0TICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	SC0R IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Serial 0 Reception End Interrupt Enable Flag**	0: Disable 1: Enable

SC1TICL : x'00FC94'

Serial 1 Transmission End Interrupt Control Register

8-bit access register

SC1TICL requests and verifies a serial 1 transmission end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	SC1T IR	-	-	-	SC1T ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Serial 1 Transmission End Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Serial 1 Transmission End Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

SC1TICH : x'00FC95'

Serial 1 Transmission End Interrupt Control Register

8-bit access register

SC1TICH enables a seial 1 transmission end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the SC0TLV[2:0] bits of the SC0TICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	SC1T IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Serial 1 Transmission End Interrupt Enable Flag**	0: Disable 1: Enable

S

SC1RICL : x'00FC96'

Serial 1 Reception End Interrupt Control Register

8-bit access register

SC1RICL requests and verifies a seial1 reception end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	SC1R IR	-	-	-	SC1R ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Serial 1 Reception End Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Serial 1 Reception End Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

SC1RICH : x'00FC97'

Serial 1 Reception End Interrupt Control Register

8-bit access register

SC1RICH enables a serial 1 reception end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the SC0TLV[2:0] bits of the SC0TICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	SC1R IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Serial 1 Reception End Interrupt Enable Flag**	0: Disable 1: Enable

SC2TICL : x'00FC98'

Serial 2 Transmission End Interrupt Control Register

8-bit access register

SC2TICL requests and verifies a serial 2 transmission end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	SC2T IR	-	-	-	SC2T ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

Bit	Name	Description
4	**Serial 2 Transmission End Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Serial 2 Transmission End Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

SC2TICH : x'00FC99'

Serial 2 Transmission End Interrupt Control Register

8-bit access register

SC2TICH sets a seial 2 transmission end interrupt level and enables an interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	SC2T LV2	SC2T LV1	SC2T LV0	-	-	-	SC2T IE
R	R/W	R/W	R/W	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0/1	0/1	0/1	0	0	0	0/1

Bit	Name	Description
6:4	**Serial 2 Transmission End Interrupt Level Setup**	Set the level from 0 to 6
0	**Serial 2 Transmission End Interrupt Enable Flag**	0: Disable 1: Enable

S

SC2RICL : x'00FC9A'

Serial 2 Reception End Interrupt Control Register

8-bit access register

SC2RICL requests and verifies a seial 2 reception end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	SC2R IR	-	-	-	SC2R ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

Bit	Name	Setting
4	**Serial 2 Reception End Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Serial 2 Reception End Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

SC2RICH : x'00FC9B'

Serial 2 Reception End Interrupt Control Register

8-bit access register

SC2RICH enables a serial 2 reception end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the SC2TLV[2:0] bits of the SC2TICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	SC2R IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

Bit	Name	Setting
0	**Serial 2 Reception End Interrupt Enable Flag**	0: Disable 1: Enable

SC3TICL : x'00FC9C'

Serial 3 Transmission End Interrupt Control Register

8-bit access register

SC3TICL requests and verifies a serial 3 transmission end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	SC3T IR	-	-	-	SC3T ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4 **Serial 3 Transmission End Interrupt Request Flag**
0: No interrupt requested
1: Interrupt requested

0 **Serial 3 Transmission End Interrupt Detect Flag**
0: Interrupt undetected
1: Interrupt detected

SC3TICH : x'00FC9D'

Serial 3 Transmission End Interrupt Control Register

8-bit access register

SC3TICH enables a seial 3 transmission end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the SC2TLV[2:0] bits of the SC2TICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	SC3T IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0 **Serial 3 Transmission End Interrupt Enable Flag**
0: Disable
1: Enable

SC3RICL : x'00FC9E'

Serial 3 Reception End Interrupt Control Register

8-bit access register

SC3RICL requests and verifies a seial 3 reception end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	SC3R IR	-	-	-	SC3R ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Serial 3 Reception End Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Serial 3 Reception End Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

SC3RICH : x'00FC9F'

Serial 3 Reception End Interrupt Control Register

8-bit access register

SC3RICH enables a serial 3 reception end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the SC2TLV[2:0] bits of the SC2TICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	SC3R IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Serial 3 Reception End Interrupt Enable Flag**	0: Disable 1: Enable

SC4TICL : x'00FCA0'

Serial 4 Transmission End Interrupt Control Register

8-bit access register

SC4TICL requests and verifies a serial 4 transmission end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	SC4T IR	-	-	-	SC4T ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Serial 4 Transmission End Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Serial 4 Transmission End Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

SC4TICH : x'00FCA1'

Serial 4 Transmission End Interrupt Control Register

8-bit access register

SC4TICH sets a seial 4 transmission end interrupt level and enables an interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	SC4T LV2	SC4T LV1	SC4T LV0	-	-	-	SC4T IE
R	R/W	R/W	R/W	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0/1	0/1	0/1	0	0	0	0/1

6:4	**Serial 4 Transmission End Interrupt Level Setup**	Set the level from 0 to 6
0	**Serial 4 Transmission End Interrupt Enable Flag**	0: Disable 1: Enable

S

SC4RICL : x'00FCA2'

Serial 4 Reception End Interrupt Control Register

8-bit access register

SC4RICL requests and verifies a seial 4 reception end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	SC4RIR	-	-	-	SC4RID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**Serial 4 Reception End Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**Serial 4 Reception End Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

SC4RICH : x'00FCA3'

Serial 4 Reception End Interrupt Control Register

8-bit access register

SC4RICH enables a serial 4 reception end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the SC4TLV[2:0] bits of the SC4TICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	SC4RIE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**Serial 4 Reception End Interrupt Enable Flag**	0: Disable 1: Enable

ETC0ICL : x'00FCA4'

ETC 0 Transfer End Interrupt Control Register

8-bit access register

ETC0ICL requests and verifies a ETC 0 transfer end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	ETC0 IR	-	-	-	ETC0 ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**ETC 0 Transfer End Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**ETC 0 Transfer End Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

E

ETC0ICH : x'00FCA5'

ETC 0 Transfer End Interrupt Control Register

8-bit access register

ETC0ICH enables a ETC 0 transfer end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the SC4TLV[2:0] bits of the SC4TICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	ETC0 IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**ETC 0 Transfer End Interrupt Enable Flag**	0: Disable 1: Enable

S

ETC1ICL : x'00FCA6'

ETC 1 Transfer End Interrupt Control Register

8-bit access register

ETC1ICL requests and verifies a ETC 1 transfer end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	ETC1 IR	-	-	-	ETC1 ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

Bit	Name	Setting
4	**ETC 1 Transfer End Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**ETC 1 Transfer End Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

ETC1ICH : x'00FCA7'

ETC 1 Transfer End Interrupt Control Register

8-bit access register

ETC1ICH enables a ETC 1 transfer end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the SC4TLV[2:0] bits of the SC4TICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	ETC1 IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

Bit	Name	Setting
0	**ETC 1 Transfer End Interrupt Enable Flag**	0: Disable 1: Enable

AT0ICL : x'00FCA8'

ATC 0 Transfer End Interrupt Control Register

8-bit access register

AT0ICL requests and verifies an ATC 0 transfer end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	AT0 IR	-	-	-	AT0 ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**ATC 0 Transfer End Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**ATC 0 Transfer End Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

AT0ICH : x'00FCA9'

ATC 0 Transfer End Interrupt Control Register

8-bit access register

AT0ICH sets an ATC 0 transfer end interrupt level and enables an interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	AT0 LV2	AT0 LV1	AT0 LV0	-	-	-	AT0 IE
R	R/W	R/W	R/W	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0/1	0/1	0/1	0	0	0	0/1

6:4	**ATC 0 Transfer End Interrupt Level Setup**	Set the level from 0 to 6
0	**ATC 0 Transfer End Interrupt Enable Flag**	0: Disable 1: Enable

AT1ICL : x'00FCAA'

ATC 1 Transfer End Interrupt Control Register

8-bit access register

AT1ICL requests and verifies an ATC 1 transfer end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	AT1 IR	-	-	-	AT1 ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**ATC 1 Transfer End Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**ATC 1 Transfer End Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

AT1ICH : x'00FCAB'

ATC 1 Transfer End Interrupt Control Register

8-bit access register

AT1ICH enables an ATC 1 transfer end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the AT0LV[2:0] bits of the AT0ICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	AT1 IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**ATC 1 Transfer End Interrupt Enable Flag**	0: Disable 1: Enable

AT2ICL : x'00FCAC'

ATC 2 Transfer End Interrupt Control Register

8-bit access register

AT2ICL requests and verifies an ATC 2 transfer end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	AT2 IR	-	-	-	AT2 ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**ATC 2 Transfer End Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**ATC 2 Transfer End Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

AT2ICH : x'00FCAD'

ATC 2 Transfer End Interrupt Control Register

8-bit access register

AT2ICH enables an ATC 2 transfer end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the AT0LV[2:0] bits of the AT0ICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	AT2 IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**ATC 2 Transfer End Interrupt Enable Flag**	0: Disable 1: Enable

AT3ICL : x'00FCAE'

ATC 3 Transfer End Interrupt Control Register

8-bit access register

AT3ICL requests and verifies an ATC 3 transfer end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	AT3 IR	-	-	-	AT3 ID
R	R	R	R/W	R	R	R	R
0	0	0	0	0	0	0	0
0	0	0	0/1	0	0	0	0/1

4	**ATC 3 Transfer End Interrupt Request Flag**	0: No interrupt requested 1: Interrupt requested
0	**ATC 3 Transfer End Interrupt Detect Flag**	0: Interrupt undetected 1: Interrupt detected

AT3ICH : x'00FCAF'

ATC 3 Transfer End Interrupt Control Register

8-bit access register

AT3ICH enables an ATC 3 transfer end interrupt.

This register allows only byte-accesses. Use the MOVB instruction to set the data. The interrupt level is the same level set in the AT0LV[2:0] bits of the AT0ICH register.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	AT3 IE
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

0	**ATC 3 Transfer End Interrupt Enable Flag**	0: Disable 1: Enable

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	IQ4 TG1	IQ4 TG0	IQ3 TG1	IQ3 TG0	IQ2 TG1	IQ2 TG0	IQ1 TG1	IQ1 TG0	IQ0 TG1	IQ0 TG0
R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

IRQTRG : x'00FCB0'

External Interrupt Condition Setup Register

16-bit access register

IRQTRG sets the trigger conditions for external interrupts.

9:8 Set Trigger Conditions for $\overline{\text{IRQ4}}$ Pin Interrupt

7:6 Set Trigger Conditions for $\overline{\text{IRQ3}}$ Pin Interrupt

5:4 Set Trigger Conditions for $\overline{\text{IRQ2}}$ Pin Interrupt

3:2 Set Trigger Conditions for $\overline{\text{IRQ1}}$ Pin Interrupt

1:0 Set Trigger Conditions for $\overline{\text{IRQ0}}$ Pin Interrupt

TG1	TG0	Trigger Condition
0	0	Low Level
0	1	Both Edges (Positive edge, Negative edge)
1	0	Falling edge (Negative edge)
1	1	Rising edge (Positive edge)

KEYTRG : x'00FCB2'

External Key Interrupt Condition Setup Register

16-bit access register

KEYTRG sets the trigger conditions for external key interrupts.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
KI7 TG1	KI7 TG0	KI6 TG1	KI6 TG0	KI5 TG1	KI5 TG0	KI4 TG1	KI4 TG0	KI3 TG1	KI3 TG0	KI2 TG1	KI2 TG0	KI1 TG1	KI1 TG0	KI0 TG1	KI0 TG0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

15:14 Set Trigger Conditions for $\overline{\text{KI7}}$ Pin Interrupt

13:12 Set Trigger Conditions for $\overline{\text{KI6}}$ Pin Interrupt

11:10 Set Trigger Conditions for $\overline{\text{KI5}}$ Pin Interrupt

9:8 Set Trigger Conditions for $\overline{\text{KI4}}$ Pin Interrupt

7:6 Set Trigger Conditions for $\overline{\text{KI3}}$ Pin Interrupt

5:4 Set Trigger Conditions for $\overline{\text{KI2}}$ Pin Interrupt

3:2 Set Trigger Conditions for $\overline{\text{KI1}}$ Pin Interrupt

1:0 Set Trigger Conditions for $\overline{\text{KI0}}$ Pin Interrupt

TG1	TG0	Trigger Condition
0	0	Low Level
0	1	Both Edges (Positive edge, Negative edge)
1	0	Falling edge (Negative edge)
1	1	Rising edge (Positive edge)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	KI7 EN	KI6 EN	KI5 EN	KI4 EN	KI3 EN	KI2 EN	KI1 EN	KI0 EN
R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

KEYCTR : x'00FCB4'

External Key Interrupt Enable Register

16-bit access register

KEYCTR enables an external key interrupt.

When OR pin is selected, a key interrupt is generated by triggering the condition set in the KEYTRG register.

7 **Set OR Pin for $\overline{\text{KI7}}$ Pin** 0: Don't set
1: Set

6 **Set OR Pin for $\overline{\text{KI6}}$ Pin** 0: Don't set
1: Set

5 **Set OR Pin for $\overline{\text{KI5}}$ Pin** 0: Don't set
1: Set

4 **Set OR Pin for $\overline{\text{KI4}}$ Pin** 0: Don't set
1: Set

3 **Set OR Pin for $\overline{\text{KI3}}$ Pin** 0: Don't set
1: Set

2 **Set OR Pin for $\overline{\text{KI2}}$ Pin** 0: Don't set
1: Set

1 **Set OR Pin for $\overline{\text{KI1}}$ Pin** 0: Don't set
1: Set

0 **Set OR Pin for $\overline{\text{KI0}}$ Pin** 0: Don't set
1: Set

K

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WD CLR	-	-	-	-	WD P2	WD P1	WD P0	-	-	-	-	-	-	-	WD RST
R/W	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0	0	0	0	0/1	0/1	0/1	0	0	0	0	0	0	0	0/1

WDREG : x'00FCB6'

Watchdog Interrupt Extension Control Register

16-bit access register

WDREG extends the watchdog interrupt cycles set in the CPUM register.

The extended watchdog counter count during those setting time.

15 Expansion Watchdog Counter Clear

0: Don't clear
1: Clear

10:8 Watchdog Interrupt Generation Time

000: Watchdog time set in CPUM register × 1
001: Watchdog time set in CPUM register × 4
010: Watchdog time set in CPUM register × 16
011: Watchdog time set in CPUM register × 64
100: Watchdog time set in CPUM register × 256
101: Watchdog time set in CPUM register × 1024
110: Watchdog time set in CPUM register × 4096

0 Watchdog Timer Reset

0: Don't reset
1: Reset

7	6	5	4	3	2	1	0
SYS C7	SYS C6	SYS C5	SYS C4	SYS C3	SYS C2	SYS C1	SYS C0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	1	1	1	1	1	0	1
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

SYSCTL : x'00FCD0'

System Control Register

8-bit access register

SYSCTL disables programming of registers related to system control.

Programming registers related to system control is disabled by writing the value except x'7D' to the SYSCTL register. This prevents programming these registers when the CPU runs erroneous operations.

7:0 Programming Disable of Registers Related to System Operations

7D: Enable all register programming
Others: Disable programming for the following registers

CPU Control	CPUM, EFCR
Address Break	ADB0, ADB1 ADBCTL
Memory Control	EXWMD MEMMD1 MEMMD2 DRAMMD1 DRAMMD2
Ports	P0MD, P1LMD, P1HMD P1MD, P3LMD, P3HMD P4LMD, P4HMD P5LMD, P5HMD, P6MD P7LMD, P7HMD P8LMD, P8MMD, P8HMD P9LMD, P9HMD PAMD, PBMD

S

W

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADB0 A15	ADB0 A14	ADB0 A13	ADB0 A12	ADB0 A11	ADB0 A10	ADB0 A9	ADB0 A8	ADB0 A7	ADB0 A6	ADB0 A5	ADB0 A4	ADB0 A3	ADB0 A2	ADB0 A1	ADB0 A0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	ADB0 A23	ADB0 A22	ADB0 A21	ADB0 A20	ADB0 A19	ADB0 A18	ADB0 A17	ADB0 A16
R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

ADB0 : x'00FCD2'

Address Break 0 Address Pointer

16/24-bit access register

ADB0 sets the address to stop address break 0 operation.

This register writes only 24-bit data or 16-bit data. Use the MOV instruction or the MOVX instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ADB1 A15	ADB1 A14	ADB1 A13	ADB1 A12	ADB1 A11	ADB1 A10	ADB1 A9	ADB1 A8	ADB1 A7	ADB1 A6	ADB1 A5	ADB1 A4	ADB1 A3	ADB1 A2	ADB1 A1	ADB1 A0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	ADB1 A23	ADB1 A22	ADB1 A21	ADB1 A20	ADB1 A19	ADB1 A18	ADB1 A17	ADB1 A16
R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

ADB1 : x'00FCD6'

Address Break 1 Address Pointer

16/24-bit access register

ADB1 sets the address to stop address break 1 operation.

This register writes only 24-bit data or 16-bit data. Use the MOV instruction or the MOVX instruction to set the data.

7	6	5	4	3	2	1	0
-	-	-	-	ADB1 ON	ADB0 ON	ADB1 CK	ADB0 CK
R	R	R	R	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0/1	0/1	0/1	0/1

ADBCTL : x'00FCDA'

Address Break Control Register

8-bit access register

ADBCTL selects the address break function and verifies the address break is generated.

3 **Address Break 1 On/Off**
0: Off
1: On

2 **Address Break 0 On/Off**
0: Off
1: On

1 **Address Break 1 Generation**
0: Not generated
1: Generated

0 **Address break 0 Generation**
0: Not generated
1: Generated

AT0CTR : x'00FD00'

ATC 0 Control Register

16-bit access register

AT0CTR sets the ATC0 operating control conditions.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AT0 EN	AT0 MD1	AT0 MD0	AT0 BW	AT0 DB8	AT0 DI	AT0 SB8	AT0 SI	-	-	-	-	AT0 IQ3	AT0 IQ2	AT0 IQ1	AT0 IQ0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0	0	0	0	0/1	0/1	0/1	0/1

15 Transfer Busy/Start Flag
0: Disable
1: Transfer start/transfer in progress

14:13 Transfer Mode
00: One byte/word transfer
01: Burst transfer
10: Two bytes transfer
11: Reserved

Selecting the two bytes transfer mode is valid only in byte access. The LSB of the address in the first byte forcibly becomes 0, and the LSB of the address in the second byte forcibly becomes 1.

12 Transfer Units
0: Word
1: Byte

Selecting word as the unit is not allowed when 8-bit bus width is allowed in the external memory space.

11 Destination Bus Width
0: 16-bit
1: 8-bit

10 Destination Pointer Increment
0: Fixed
1: Increment

Selecting 8-bit destination bus width or 8-bit source bus width is allowed only when 8-bit bus width is selected in the external memory space.

9 Source Bus Width
0: 16-bit
1: 8-bit

8 Source Pointer Increment
0: Fixed
1: Increment

When destination pointer increment or source pointer increment is selected, the pointer increments by 1 in byte access and by 2 in word access.

3:0 ATC Activation Factor Setup
0000: Software Initialization
0001: $\overline{\text{DMAREQ0}}$ pin input
0010: External interrupt 0
0011: External interrupt 1
0100: External interrupt 4
0101: Timer 3 underflow interrupt
0110: Timer 7 underflow interrupt
0111: Timer 9 underflow interrupt
1000: Timer 10 capture A interrupt
1001: Timer 11 capture B interrupt
1010: Serial 0 transmission end interrupt
1011: Serial 0 reception end interrupt
1100: Serial 3 transmission end interrupt
1101: Serial 3 reception end interrupt
1110: A/D conversion end interrupt
1111: Key interrupt

The AT0IQ[3:0] bits are cleared to 0 by the ATC0 transfer end interrupt.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	AT0 CNT11	AT0 CNT10	AT 0 CNT9	AT0 CNT8	AT0 CNT7	AT0 CNT6	AT0 CNT5	AT0 CNT4	AT0 CNT3	AT0 CNT2	AT0 CNT1	AT0 CNT0
R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AT0CNT : x'00FD02'

ATC 0 Transfer Word Count Register

16-bit access register

AT0CNT sets the bytes to be transferred subtracted by 1. Decrement by 1 every time 1-byte data is transferred and reach x'0FFF' when the transfer is completed.

This register writes only 16-bit data. Use the MOV instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AT0 SRC15	AT0 SRC14	AT0 SRC13	AT0 SRC12	AT0 SRC11	AT0 SRC10	AT0 SRC9	AT0 SRC8	AT0 SRC7	AT0 SRC6	AT0 SRC5	AT0 SRC4	AT0 SRC3	AT0 SRC2	AT0 SRC1	AT0 SRC0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	AT0 SRC23	AT0 SRC22	AT0 SRC21	AT0 SRC20	AT0 SRC19	AT0 SRC18	AT0 SRC17	AT0 SRC16
R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AT0SRC : x'00FD04'

ATC 0 Source Address Pointer

16/24-bit access register

AT0SRC sets the transfer source address. When the source pointer increment bit is set to be fixed, the transfer source address do not change. When the source pointer increment bit is set to increment, increment by 1 in the byte transfer and by 2 in the word transfer.

This register writes only 24-bit data or 16-bit data. Use the MOV instruction or the MOVX instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AT0 DST15	AT0 DST14	AT0 DST13	AT0 DST12	AT0 DST11	AT0 DST10	AT0 DST9	AT0 DST8	AT0 DST7	AT0 DST6	AT0 DST5	AT0 DST4	AT0 DST3	AT0 DST2	AT0 DST1	AT0 DST0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	AT0 DST23	AT0 DST22	AT0 DST21	AT0 DST20	AT0 DST19	AT0 DST18	AT0 DST17	AT0 DST16
R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AT0DST : x'00FD08'

ATC 0 Destination Address Pointer

16/24-bit access register

AT0DST sets the transfer destination address. When the destination pointer increment bit is set to be fixed, the transfer source address do not change. When the source pointer increment bit is set to increment, increment by 1 in the byte transfer and by 2 in the word transfer.

This register writes only 24-bit data or 16-bit data. Use the MOV instruction or the MOVX instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AT1 EN	AT1 MD1	AT1 MD0	AT1 BW	AT1 DB8	AT1 DI	AT1 SB8	AT1 SI	-	-	-	-	AT1 IQ3	AT1 IQ2	AT1 IQ1	AT1 IQ0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0	0	0	0	0/1	0/1	0/1	0/1

AT1CTR : x'00FD10'

ATC 1 Control Register

16-bit access register

AT1CTR sets the ATC1 operating control conditions.

Selecting the two bytestransfer mode is valid only in byte access. The LSB of the address in the first byte forcibly becomes 0, and the LSB of the address in the second byte forcibly becomes 1.

Selecting word as the unit is not allowed when 8-bit bus width is allowed in the external memory space.

Selecting 8-bit destination bus width or 8-bit source bus width is allowed only when 8-bit bus width is selected in the external memory space.

When destination pointer increment or source pointer increment is selected, the pointer increments by 1 in byte access and by 2 in word access.

The AT1IQ[3:0] bits are cleared to 0 by the ATC1 transfer end interrupt.

15 Transfer Busy/Start Flag
0: Disable
1: Transfer start/transfer in progress

14:13 Transfer Mode
00: One byte/word transfer
01: Burst transfer
10: Two bytes transfer
11: Reserved

12 Transfer Units
0: Word
1: Byte

11 Destination Bus Width
0: 16-bit
1: 8-bit

10 Destination Pointer Increment
0: Fixed
1: Increment

9 Source Bus Width
0: 16-bit
1: 8-bit

8 Source Pointer Increment
0: Fixed
1: Increment

3:0 ATC Activation Factor Setup
0000: Software Initialization
0001: $\overline{\text{DMAREQ1}}$ pin input
0010: External interrupt 2
0011: External interrupt 3
0100: Timer 0 underflow interrupt
0101: Timer 4 underflow interrupt
0110: Timer 8 underflow interrupt
0111: Timer 9 capture A interrupt
1000: Timer 10 capture B interrupt
1001: Timer 12 capture B interrupt
1010: Serial 0 transmission end interrupt
1011: Serial 0 reception end interrupt
1100: Serial 1 transmission end interrupt
1101: Serial 1 reception end interrupt
1110: Serial 4 transmission end interrupt
1111: Serial 4 reception end interrupt

AT1CNT : x'00FD12'

ATC 1 Transfer Word Count Register

16-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	AT1 CNT11	AT1 CNT10	AT1 CNT9	AT1 CNT8	AT1 CNT7	AT1 CNT6	AT1 CNT5	AT1 CNT4	AT1 CNT3	AT1 CNT2	AT1 CNT1	AT1 CNT0
R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AT1CNT sets the bytes to be transferred subtracted by 1. Decrement by 1 every time 1-byte data is transferred and reach x'0FFF' when the transfer is completed.

This register writes only 16-bit data. Use the MOV instruction to set the data.

AT1SRC : x'00FD14'

ATC 1 Source Address Pointer

16/24-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AT1 SRC15	AT1 SRC14	AT1 SRC13	AT1 SRC12	AT1 SRC11	AT1 SRC10	AT1 SRC9	AT1 SRC8	AT1 SRC7	AT1 SRC6	AT1 SRC5	AT1 SRC4	AT1 SRC3	AT1 SRC2	AT1 SRC1	AT1 SRC0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	AT1 SRC23	AT1 SRC22	AT1 SRC21	AT1 SRC20	AT1 SRC19	AT1 SRC18	AT1 SRC17	AT1 SRC16
R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AT1SRC sets the transfer source address. When the source pointer increment bit is set to be fixed, the transfer source address do not change. When the source pointer increment bit is set to increment, increment by 1 in the byte transfer and by 2 in the word transfer.

This register writes only 24-bit data or 16-bit data. Use the MOV instruction or the MOVX instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AT1 DST15	AT1 DST14	AT1 DST13	AT1 DST12	AT1 DST11	AT1 DST10	AT1 DST9	AT1 DST8	AT1 DST7	AT1 DST6	AT1 DST5	AT1 DST4	AT1 DST3	AT1 DST2	AT1 DST1	AT1 DST0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	AT1 DST23	AT1 DST22	AT1 DST21	AT1 DST20	AT1 DST19	AT1 DST18	AT1 DST17	AT1 DST16
R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AT1DST : x'00FD18'

ATC 1 Destination Address Pointer

16/24-bit access register

AT1DST sets the transfer destination address. When the destination pointer increment bit is set to be fixed, the transfer source address do not change. When the source pointer increment bit is set to increment, increment by 1 in the byte transfer and by 2 in the word transfer.

This register writes only 24-bit data or 16-bit data. Use the MOV instruction or the MOVX instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AT2 EN	AT2 MD1	AT2 MD0	AT2 BW	AT2 DB8	AT2 DI	AT2 SB8	AT2 SI	-	-	-	-	AT2 IQ3	AT2 IQ2	AT2 IQ1	AT2 IQ0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0	0	0	0	0/1	0/1	0/1	0/1

AT2CTR : x'00FD20'

ATC 2 Control Register

16-bit access register

AT2CTR sets the ATC2 operating control conditions.

Selecting the two bytes transfer mode is valid only in byte access. The LSB of the address in the first byte forcibly becomes 0, and the LSB of the address in the second byte forcibly becomes 1.

Selecting word as the unit is not allowed when 8-bit bus width is allowed in the external memory space.

Selecting 8-bit destination bus width or 8-bit source bus width is allowed only when 8-bit bus width is selected in the external memory space.

When destination pointer increment or source pointer increment is selected, the pointer increments by 1 in byte access and by 2 in word access.

The AT2IQ[3:0] bits are cleared to 0 by the ATC2 transfer end interrupt.

15 Transfer Busy/Start Flag
0: Disable
1: Transfer start/transfer in progress

14:13 Transfer Mode
00: One byte/word transfer
01: Burst transfer
10: Two bytes transfer
11: Reserved

12 Transfer Units
0: Word
1: Byte

11 Destination Bus Width
0: 16-bit
1: 8-bit

10 Destination Pointer Increment
0: Fixed
1: Increment

9 Source Bus Width
0: 16-bit
1: 8-bit

8 Source Pointer Increment
0: Fixed
1: Increment

3:0 ATC Activation Factor Setup
0000: Software Initialization
0001: $\overline{\text{DMAREQ0}}$ pin input
0010: External interrupt 0
0011: External interrupt 1
0100: Timer 1 underflow interrupt
0101: Timer 5 underflow interrupt
0110: Timer 8 capture A interrupt
0111: Timer 9 capture B interrupt
1000: Timer 11 underflow interrupt
1001: Timer 12 capture A interrupt
1010: Serial 1 transmission end interrupt
1011: Serial 1 reception end interrupt
1100: Serial 2 transmission end interrupt
1101: Serial 2 reception end interrupt
1110: Serial 4 transmission end interrupt
1111: Serial 4 reception end interrupt

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	AT2 CNT11	AT2 CNT10	AT2 CNT9	AT2 CNT8	AT2 CNT7	AT2 CNT6	AT2 CNT5	AT2 CNT4	AT2 CNT3	AT2 CNT2	AT2 CNT1	AT2 CNT0
R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AT2CNT : x'00FD22'

ATC 2 Transfer Word Count Register

16-bit access register

AT2CNT sets the bytes to be transferred subtracted by 1. Decrement by 1 every time 1-byte data is transferred and reach x'0FFF' when the transfer is completed.

This register writes only 16-bit data. Use the MOV instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AT2 SRC15	AT2 SRC14	AT2 SRC13	AT2 SRC12	AT2 SRC11	AT2 SRC10	AT2 SRC9	AT2 SRC8	AT2 SRC7	AT2 SRC6	AT2 SRC5	AT2 SRC4	AT2 SRC3	AT2 SRC2	AT2 SRC1	AT2 SRC0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	AT2 SRC23	AT2 SRC22	AT2 SRC21	AT2 SRC20	AT2 SRC19	AT2 SRC18	AT2 SRC17	AT2 SRC16
R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AT2SRC : x'00FD24'

ATC 2 Source Address Pointer

16/24-bit access register

AT2SRC sets the transfer source address. When the source pointer increment bit is set to be fixed, the transfer source address do not change. When the source pointer increment bit is set to increment, increment by 1 in the byte transfer and by 2 in the word transfer.

This register writes only 24-bit data or 16-bit data. Use the MOV instruction or the MOVX instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AT2 DST15	AT2 DST14	AT2 DST13	AT2 DST12	AT2 DST11	AT2 DST10	AT2 DST9	AT2 DST8	AT2 DST7	AT2 DST6	AT2 DST5	AT2 DST4	AT2 DST3	AT2 DST2	AT2 DST1	AT2 DST0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	AT2 DST23	AT2 DST22	AT2 DST21	AT2 DST20	AT2 DST19	AT2 DST18	AT2 DST17	AT2 DST16
R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AT2DST : x'00FD28'

ATC 2 Destination Address Pointer

16/24-bit access register

AT2DST sets the transfer destination address. When the destination pointer increment bit is set to be fixed, the transfer source address do not change. When the source pointer increment bit is set to increment, increment by 1 in the byte transfer and by 2 in the word transfer.

This register writes only 24-bit data or 16-bit data. Use the MOV instruction or the MOVX instruction to set the data.

AT3CTR : x'00FD30'

ATC 3 Control Register

16-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AT3 EN	AT3 MD1	AT3 MD0	AT3 BW	AT3 DB8	AT3 DI	AT3 SB8	AT3 SI	-	-	-	-	AT3 IQ3	AT3 IQ2	AT3 IQ1	AT3 IQ0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0	0	0	0	0/1	0/1	0/1	0/1

15 Transfer Busy/Start Flag
0: Disable
1: Transfer start/transfer in progress

14:13 Transfer Mode
00: One byte/word transfer
01: Burst transfer
10: Two bytes transfer
11: Reserved

12 Transfer Units
0: Word
1: Byte

11 Destination Bus Width
0: 16-bit
1: 8-bit

10 Destination Pointer Increment
0: Fixed
1: Increment

9 Source Bus Width
0: 16-bit
1: 8-bit

8 Source Pointer Increment
0: Fixed
1: Increment

3:0 ATC Activation Factor Setup
0000: Software Initialization
0001: $\overline{\text{DMAREQ1}}$ pin input
0010: External interrupt 2
0011: External interrupt 3
0100: Timer 2 underflow interrupt
0101: Timer 6 underflow interrupt
0110: Timer 8 capture B interrupt
0111: Timer 10 underflow interrupt
1000: Timer 11 capture A interrupt
1001: Timer 12 capture B interrupt
1010: Serial 2 transmission end interrupt
1011: Serial 2 reception end interrupt
1100: Serial 3 transmission end interrupt
1101: Serial 3 reception end interrupt
1110: A/D conversion end interrupt
1111: Key interrupt

AT3CTR sets the ATC3 operating control conditions.

Selecting the two bytes transfer mode is valid only in byte access. The LSB of the address in the first byte forcibly becomes 0, and the LSB of the address in the second byte forcibly becomes 1.

Selecting word as the unit is not allowed when 8-bit bus width is allowed in the external memory space.

Selecting 8-bit destination bus width or 8-bit source bus width is allowed only when 8-bit bus width is selected in the external memory space.

When destination pointer increment or source pointer increment is selected, the pointer increments by 1 in byte access and by 2 in word access.

The AT3IQ[3:0] bits are cleared to 0 by the ATC3 transfer end interrupt.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	AT3 CNT11	AT3 CNT10	AT3 CNT9	AT3 CNT8	AT3 CNT7	AT3 CNT6	AT3 CNT5	AT3 CNT4	AT3 CNT3	AT3 CNT2	AT3 CNT1	AT3 CNT0
R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AT3CNT : x'00FD32'

ATC 3 Transfer Word Count Register

16-bit access register

AT3CNT sets the bytes to be transferred subtracted by 1. Decrement by 1 every time 1-byte data is transferred and reach x'0FFF' when the transfer is completed.

This register writes only 16-bit data. Use the MOV instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AT3 SRC15	AT3 SRC14	AT3 SRC13	AT3 SRC12	AT3 SRC11	AT3 SRC10	AT3 SRC9	AT3 SRC8	AT3 SRC7	AT3 SRC6	AT3 SRC5	AT3 SRC4	AT3 SRC3	AT3 SRC2	AT3 SRC1	AT3 SRC0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	AT3 SRC23	AT3 SRC22	AT3 SRC21	AT3 SRC20	AT3 SRC19	AT3 SRC18	AT3 SRC17	AT3 SRC16
R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AT3SRC : x'00FD34'

ATC 3 Source Address Pointer

16/24-bit access register

AT3SRC sets the transfer source address. When the source pointer increment bit is set to be fixed, the transfer source address do not change. When the source pointer increment bit is set to increment, increment by 1 in the byte transfer and by 2 in the word transfer.

This register writes only 24-bit data or 16-bit data. Use the MOV instruction or the MOVX instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
AT3 DST15	AT3 DST14	AT3 DST13	AT3 DST12	AT3 DST11	AT3 DST10	AT3 DST9	AT3 DST8	AT3 DST7	AT3 DST6	AT3 DST5	AT3 DST4	AT3 DST3	AT3 DST2	AT3 DST1	AT3 DST0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	AT3 DST23	AT3 DST22	AT3 DST21	AT3 DST20	AT3 DST19	AT3 DST18	AT3 DST17	AT3 DST16
R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AT3DST : x'00FD38'

ATC 3 Destination Address Pointer

16/24-bit access register

AT3DST sets the transfer destination address. When the destination pointer increment bit is set to be fixed, the transfer source address do not change. When the source pointer increment bit is set to increment, increment by 1 in the byte transfer and by 2 in the word transfer.

This register writes only 24-bit data or 16-bit data. Use the MOV instruction or the MOVX instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ET0 FLG	-	ET0 MD0	ET0 BW	ET0 DB8	ET0 DI	ET0 SB8	ET0 SI	-	-	-	-	-	-	ET0 DIR	ET0 EN
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0	0/1	0/1	0/1	0/1	0/1	0/1	0	0	0	0	0	0	0/1	0/1

ET0CTR : x'00FD40'

ETC 0 Control Register

16-bit access register

ET0CTR sets the ETC0 operating control conditions. Transfers the data automatically between the external device with ACK input function and the external memory.

15 Transfer Busy/Start Flag
0: Disable
1: Transfer start/transfer in progress

13 Transfer Mode
0: One byte/word transfer
1: Burst transfer

12 Transfer Units
0: Word
1: Byte

Selecting word as the unit is not allowed when 8-bit bus width is allowed in the external memory space.

11 Destination Bus Width
0: 16-bit
1: 8-bit

Selecting 8-bit destination bus width or 8-bit source bus width is allowed only when 8-bit bus width is selected in the external memory space.

10 Destination Pointer Increment
0: Fixed
1: Increment

9 Source Bus Width
0: 16-bit
1: 8-bit

When destination pointer increment or source pointer increment is selected, the pointer increments by 1 in byte access and by 2 in word access.

8 Source Pointer Increment
0: Fixed
1: Increment

1 Transfer Direction
0: External device → external memory
1: External memory → external device

0 ETC Transfer Enable
0: Disable
1: Enable

The ET0EN flag is cleared to 0 by the ETC0 transfer end interrupt.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	ET0 CNT11	ET0 CNT10	ET 0 CNT9	ET0 CNT8	ET0 CNT7	ET0 CNT6	ET0 CNT5	ET0 CNT4	ET0 CNT3	ET0 CNT2	ET0 CNT1	ET0 CNT0
R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

ET0CNT : x'00FD42'

ETC 0 Transfer Word Count Register

E

16-bit access register

ET0CNT sets the bytes to be transferred subtracted by 1. Decrement by 1 every time 1-byte/word data is transferred and reach x'0FFF' when the transfer is completed.

This register writes only 16-bit data. Use the MOV instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ET0 SRC15	ET0 SRC14	ET0 SRC13	ET0 SRC12	ET0 SRC11	ET0 SRC10	ET0 SRC9	ET0 SRC8	ET0 SRC7	ET0 SRC6	ET0 SRC5	ET0 SRC4	ET0 SRC3	ET0 SRC2	ET0 SRC1	ET0 SRC0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	ET0 SRC23	ET0 SRC22	ET0 SRC21	ET0 SRC20	ET0 SRC19	ET0 SRC18	ET0 SRC17	ET0 SRC16
R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

ET0SRC : x'00FD44'

ETC 0 Source Address Pointer

16/24-bit access register

ET0SRC sets the transfer source address. When the source pointer increment bit is set to be fixed, the transfer source address do not change. When the source pointer increment bit is set to increment, increment by 1 in the byte transfer and by 2 in the word transfer.

This register writes only 24-bit data or 16-bit data. Use the MOV instruction or the MOVX instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ET0 DST15	ET0 DST14	ET0 DST13	ET0 DST12	ET0 DST11	ET0 DST10	ET0 DST9	ET0 DST8	ET0 DST7	ET0 DST6	ET0 DST5	ET0 DST4	ET0 DST3	ET0 DST2	ET0 DST1	ET0 DST0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	ET0 DST23	ET0 DST22	ET0 DST21	ET0 DST20	ET0 DST19	ET0 DST18	ET0 DST17	ET0 DST16
R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

ET0DST : x'00FD48'

ETC 0 Destination Address Pointer

16/24-bit access register

ET0DST sets the transfer destination address. When the destination pointer increment bit is set to be fixed, the transfer source address do not change. When the source pointer increment bit is set to increment, increment by 1 in the byte transfer and by 2 in the word transfer.

This register writes only 24-bit data or 16-bit data. Use the MOV instruction or the MOVX instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ET1 FLG	-	ET1 MD0	ET1 BW	ET1 DB8	ET1 DI	ET1 SB8	ET1 SI	-	-	-	-	-	-	ET1 DIR	ET1 EN
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0	0/1	0/1	0/1	0/1	0/1	0/1	0	0	0	0	0	0	0/1	0/1

ET1CTR : x'00FD50'

ETC 1 Control Register

E

16-bit access register

ET1CTR sets the ETC1 operating control conditions. Transfers the data automatically between the external device with ACK input function and the external memory.

15 Transfer Busy/Start Flag
0: Disable
1: Transfer start/transfer in progress

13 Transfer Mode
0: One byte/word transfer
1: Burst transfer

12 Transfer Units
0: Word
1: Byte

Selecting word as the unit is not allowed when 8-bit bus width is allowed in the external memory space.

11 Destination Bus Width
0: 16-bit
1: 8-bit

Selecting 8-bit destination bus width or 8-bit source bus width is allowed only when 8-bit bus width is selected in the external memory space.

10 Destination Pointer Increment
0: Fixed
1: Increment

9 Source Bus Width
0: 16-bit
1: 8-bit

When destination pointer increment or source pointer increment is selected, the pointer increments by 1 in byte access and by 2 in word access.

8 Source Pointer Increment
0: Fixed
1: Increment

1 Transfer Direction
0: External device → external memory
1: External memory → external device

0 ETC Transfer Enable
0: Disable
1: Enable

The ET1EN flag is cleared to 0 by the ETC1 transfer end interrupt.

ET1CNT : x'00FD52'

ETC 1 Transfer Word Count Register

16-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	ET1 CNT11	ET1 CNT10	ET1 CNT9	ET1 CNT8	ET1 CNT7	ET1 CNT6	ET1 CNT5	ET1 CNT4	ET1 CNT3	ET1 CNT2	ET1 CNT1	ET1 CNT0
R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

ET1CNT sets the bytes to be transferred subtracted by 1. Decrement by 1 every time 1-byte/word data is transferred and reach x'0FFF' when the transfer is completed.

This register writes only 16-bit data. Use the MOV instruction to set the data.

ET1SRC : x'00FD54'

ETC 1 Source Address Pointer

16/24-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ET1 SRC15	ET1 SRC14	ET1 SRC13	ET1 SRC12	ET1 SRC11	ET1 SRC10	ET1 SRC9	ET1 SRC8	ET1 SRC7	ET1 SRC6	ET1 SRC5	ET1 SRC4	ET1 SRC3	ET1 SRC2	ET1 SRC1	ET1 SRC0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	ET1 SRC23	ET1 SRC22	ET1 SRC21	ET1 SRC20	ET1 SRC19	ET1 SRC18	ET1 SRC17	ET1 SRC16
R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

ET1SRC sets the transfer source address. When the source pointer increment bit is set to be fixed, the transfer source address do not change. When the source pointer increment bit is set to increment, increment by 1 in the byte transfer and by 2 in the word transfer.

This register writes only 24-bit data or 16-bit data. Use the MOV instruction or the MOVX instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ET1 DST15	ET1 DST14	ET1 DST13	ET1 DST12	ET1 DST11	ET1 DST10	ET1 DST9	ET1 DST8	ET1 DST7	ET1 DST6	ET1 DST5	ET1 DST4	ET1 DST3	ET1 DST2	ET1 DST1	ET1 DST0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	ET1 DST23	ET1 DST22	ET1 DST21	ET1 DST20	ET1 DST19	ET1 DST18	ET1 DST17	ET1 DST16
R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	undefined	undefined	undefined	undefined	undefined	undefined	undefined	undefined
0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

ET1DST : x'00FD58'

ETC 1 Destination Address Pointer

16/24-bit access register

ET1DST sets the transfer destination address. When the destination pointer increment bit is set to be fixed, the transfer source address do not change. When the source pointer increment bit is set to increment, increment by 1 in the byte transfer and by 2 in the word transfer.

This register writes only 24-bit data or 16-bit data. Use the MOV instruction or the MOVX instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SC0 TEN	SC0 REN	SC0 BRE	reserv ed	reserv ed	-	SC0 OD	reserv ed	SC0 LN	SC0 PTY2	SC0 PTY1	SC0 PTY0	SC0 SB	-	SC0 S1	SC0 S0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0	0/1	0/1

SC0CTR : x'00FD80'

Serial 0 Control Register

16-bit access register

SC0CTR sets serial 0 operating conditions.

15 **Transmit Enable**
0: Disable
1: Enable

14 **Receive Enable**
0: Disable
1: Enable

13 **Break Transmission**
0: Don't break
1: Break

12 **Reserved**
Set to 0

11 **Reserved**
Set to 0

9 **Bit Order Selection**
0: LSB first
1: MSB first (select only when the character length is 8-bit.)

8 **Reserved**
Set to 0

7 **Character Length**
0: 7-bit
1: 8-bit

6:4 **Parity Bit Selection**
000: None
100: 0 (output low)
101: 1 (output high)
110: Even (1s are even)
111: Odd (1s are odd)
Others: Reserved

3 **SBO0 Output Hold Time**
0: More than BOSC cycles
1: More than timer 1 underflow cycles

1:0 **Serial 0 Clock Source Selection**
00: SBT0 pin
01: Timer 1 underflow (1/8)
10: Timer 2 underflow (1/2)
11: Timer 2 underflow (1/8)

The SBO0 output hold time is extended only when SBT0 pin is selected as serial 0 clock source.

SC0TRB : x'00FD82'

Serial 0 Transmit/ Receive Buffer

8-bit access register

SC0TRB writes the serial 0 transmit data and reads the serial 0 receive data.

Transmission starts by writing the data into this register. The data is received by reading this register. In 7-bit transfer, the MSB (bit 7) becomes 0. The data is read when an interrupt occurs or the SC0RXA flag of the SC0STR register is 1.

7	6	5	4	3	2	1	0
SC0 TRB7	SC0 TRB6	SC0 TRB5	SC0 TRB4	SC0 TRB3	SC0 TRB2	SC0 TRB1	SC0 TRB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 Serial Transmit/Receive Data

S

7	6	5	4	3	2	1	0
SC0 TBY	SC0 RBY	reserv ed	SC0 RXA	reserv ed	reserv ed	SC0 PE	SC0 OE
R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

SC0STR : x'00FD83'

Serial 0 Status Register

8-bit access register (16-bit access is posible from even address)

SC0STR reads the status for serial interface 0.

7 Transmission Busy Flag
0: Ready to transmit
1: Transmission in progress

6 Reception Busy Flag
0: Ready to receive
1: Reception in progress

5 Reserved

4 Received Data
0: No received data
1: Received data

3:2 Reserved

1 Parity Error
0: No error
1: Error

0 Overrun Error
0: No error
1: Error

A parity error occurs when the parity bit is 1 although it is set to 0, when the parity bit is 0 although it is set to 1, when the parity bit is odd although it is set to even, and when the parity bit is even although it is set to odd. Parity error data is updated whenever the parity bit is received.

An overrun error occurs when the next data is received completely before the CPU reads the received data (SC0TRB). Overrun error data is updated whenever the last data bit (seventh or eighth bit) is received.

Do not use the SC0RBY flag to set polling for the received data wait in clock synchronous mode. Use the interrupt service routine, the serial interrupt flag or the SC0RXA flag.

SC1CTR : x'00FD88'

Serial 1 Control Register

16-bit access register

SC1CTR sets serial 1 operating conditions.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SC1 TEN	SC1 REN	SC1 BRE	reserv ed	reserv ed	-	SC1 OD	reserv ed	SC1 LN	SC1 PTY2	SC1 PTY1	SC1 PTY0	SC1 SB	-	SC1 S1	SC1 S0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0	0/1	0/1

15 Transmit Enable
0: Disable
1: Enable

14 Receive Enable
0: Disable
1: Enable

13 Break Transmission
0: Don't break
1: Break

12 Reserved
Set to 0

11 Reserved
Set to 0

9 Bit Order Selection
0: LSB first
1: MSB first (select only when the character length is 8-bit.)

8 Reserved
Set to 0

7 Character Length
0: 7-bit
1: 8-bit

6:4 Parity Bit Selection
000: None
100: 0 (output low)
101: 1 (output high)
110: Even (1s are even)
111: Odd (1s are odd)
Others: Reserved

3 SBO1 Output Hold Time
0: More than BOSC cycles
1: More than timer 1 underflow cycles

1:0 Serial 1 Clock Source Selection
00: SBT1 pin
01: Timer 1 underflow (1/8)
10: Timer 2 underflow (1/2)
11: Timer 2 underflow (1/8)

The SBO1 output hold time is extended only when SBT1 pin is selected as serial 1 clock source.

S

SC1TRB : x'00FD8A'

Serial 1 Transmit/ Receive Buffer

8-bit access register

SC1TRB writes the serial 1 transmit data and reads the serial 1 receive data.

Transmission starts by writing the data into this register. The data is received by reading this register. In 7-bit transfer, the MSB (bit 7) becomes 0. The data is read when an interrupt occurs or the SC1RXA flag of the SC1STR register is 1.

7	6	5	4	3	2	1	0
SC1 TRB7	SC1 TRB6	SC1 TRB5	SC1 TRB4	SC1 TRB3	SC1 TRB2	SC1 TRB1	SC1 TRB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 Serial Transmit/Receive Data

SC1STR : x'00FD8B'

Serial 1 Status Register

8-bit access register (16-bit access is posible from even address)

SC1STR reads the status for serial interface 1.

7	6	5	4	3	2	1	0
SC1 TBY	SC1 RBY	reserved	SC1 RXA	reserved	reserved	SC1 PE	SC1 OE
R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7 Transmission Busy Flag
0: Ready to transmit
1: Transmission in progress

6 Reception Busy Flag
0: Ready to receive
1: Reception in progress

5 Reserved

4 Received Data
0: No received data
1: Received data

3:2 Reserved

1 Parity Error
0: No error
1: Error

0 Overrun Error
0: No error
1: Error

A parity error occurs when the parity bit is 1 although it is set to 0, when the parity bit is 0 although it is set to 1, when the parity bit is odd although it is set to even, and when the parity bit is even although it is set to odd. Parity error data is updated whenever the parity bit is received.

An overrun error occurs when the next data is received completely before the CPU reads the received data (SC1TRB). Overrun error data is updated whenever the last data bit (seventh or eighth bit) is received.

Do not poll the SC1RBY flag to verify the reception end in clock synchronous mode. Generate a serial 1 reception end interrupt or poll the SC1RXA flag to verify the reception end.

S

SC2CTR : x'00FD90'

Serial 2 Control Register

16-bit access register

SC2CTR sets serial 2 operating conditions.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SC2 TEN	SC2 REN	SC2 BRE	reserv ed	reserv ed	-	SC2 OD	reserv ed	SC2 LN	SC2 PTY2	SC2 PTY1	SC2 PTY0	SC2 SB	-	SC2 S1	SC2 S0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0	0/1	0/1

15 Transmit Enable
0: Disable
1: Enable

14 Receive Enable
0: Disable
1: Enable

13 Break Transmission
0: Don't break
1: Break

12 Reserved
Set to 0

11 Reserved
Set to 0

9 Bit Order Selection
0: LSB first
1: MSB first (select only when the character length is 8-bit.)

8 Reserved
Set to 0

7 Character Length
0: 7-bit
1: 8-bit

6:4 Parity Bit Selection
000: None
100: 0 (output low)
101: 1 (output high)
110: Even (1s are even)
111: Odd (1s are odd)
Others: Reserved

3 SBO2 Output Hold Time
0: More than BOSC cycles
1: More than timer 4 underflow cycles

1:0 Serial 2 Clock Source Selection
00: SBT2 pin
01: Timer 4 underflow (1/8)
10: Timer 5 underflow (1/2)
11: Timer 5 underflow (1/8)

The SBO2 output hold time is extended only when SBT2 pin is selected as serial 2 clock source.

SC2TRB : x'00FD92'

Serial 2 Transmit/ Receive Buffer

8-bit access register

SC2TRB writes the serial 2 transmit data and reads the serial 2 receive data.

Transmission starts by writing the data into this register. The data is received by reading this register. In 7-bit transfer, the MSB (bit 7) becomes 0. The data is read when an interrupt occurs or the SC2RXA flag of the SC2STR register is 1.

7	6	5	4	3	2	1	0
SC2 TRB7	SC2 TRB6	SC2 TRB5	SC2 TRB4	SC2 TRB3	SC2 TRB2	SC2 TRB1	SC2 TRB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 Serial Transmit/Receive Data

S

7	6	5	4	3	2	1	0
SC2 TBY	SC2 RBY	reserv ed	SC2 RXA	reserv ed	reserv ed	SC2 PE	SC2 OE
R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

SC2STR : x'00FD93'

Serial 2 Status Register

8-bit access register (16-bit access is possible from even address)

SC2STR reads the status for serial interface 2.

7 **Transmission Busy Flag**
0: Ready to transmit
1: Transmission in progress

6 **Reception Busy Flag**
0: Ready to receive
1: Reception in progress

5 **Reserved**

4 **Received Data**
0: No received data
1: Received data

3:2 **Reserved**

1 **Parity Error**
0: No error
1: Error

0 **Overrun Error**
0: No error
1: Error

A parity error occurs when the parity bit is 1 although it is set to 0, when the parity bit is 0 although it is set to 1, when the parity bit is odd although it is set to even, and when the parity bit is even although it is set to odd. Parity error data is updated whenever the parity bit is received.

An overrun error occurs when the next data is received completely before the CPU reads the received data (SC2TRB). Overrun error data is updated whenever the last data bit (seventh or eighth bit) is received.

Do not poll the SC2RBY flag to verify the reception end in clock synchronous mode. Generate a serial 2 reception end interrupt or poll the SC2RXA flag to verify the reception end.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SC3 TEN	SC3 REN	SC3 BRE	SC3 IIC	SC3 PTL	-	SC3 OD	SC3 ICM	SC3 LN	SC3 PTY2	SC3 PTY1	SC3 PTY0	SC3 SB	-	SC3 S1	SC3 S0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0	0/1	0/1

SC3CTR : x'00FD98'

Serial 3 Control Register

16-bit access register

SC3CTR sets serial 3 operating conditions.

15 Transmit Enable — 0: Disable 1: Enable

14 Receive Enable — 0: Disable 1: Enable

13 Break Transmission — 0: Don't break 1: Break

12 I²C Start or Stop Sequence
0: Stop sequence output when changing this bit from 1 to 0.
1: Start sequence output when changing this bit from 0 to 1.

11 Protocol Selection
0: Asynchronous mode
1: Clock synchronous mode, I²C mode

9 Bit Order Selection
0: LSB first
1: MSB first (select only when the character length is 8-bit.)

8 I²C mode Selection — 0: I²C mode off 1: I²C mode on

7 Character Length — 0: 7-bit 1: 8-bit

6:4 Parity Bit Selection
000: None
100: 0 (output low)
101: 1 (output high)
110: Even (1s are even)
111: Odd (1s are odd)
Others: Reserved

3 Stop Bit Selection (asynchronous mode) — 0: 1-bit 1: 2-bit

SBO3 Output Hold Time (clock synchronous mode)
0: More than BOSC cycles
1: More than timer 4 underflow cycles

The SBO3 output hold time is extended only when SBT3 pin is selected as serial 3 clock source.

1:0 Serial 3 Clock Source Selection
Asynchronous mode, I²C mode
01: Timer 4 underflow (1/8)
11: Timer 5 underflow (1/8)
Clock synchronous mode
00: SBT3 pin
01: Timer 4 underflow (1/8)
10: Timer 5 underflow (1/2)
11: Timer 5 underflow (1/8)

S

7	6	5	4	3	2	1	0
SC3 TRB7	SC3 TRB6	SC3 TRB5	SC3 TRB4	SC3 TRB3	SC3 TRB2	SC3 TRB1	SC3 TRB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

SC3TRB : x'00FD9A'

Serial 3 Transmit/ Receive Buffer

8-bit access register

SC3TRB writes the serial 3 transmit data and reads the serial 3 receive data.

Transmission starts by writing the data into this register. The data is received by reading this register. In 7-bit transfer, the MSB (bit 7) becomes 0. The data is read when an interrupt occurs or the SC3RXA flag of the SC3STR register is 1.

7:0 Serial Transmit/Receive Data

7	6	5	4	3	2	1	0
SC3 TBY	SC3 RBY	SC3 ISP	SC3 RXA	SC3 IST	SC3 FE	SC3 PE	SC3 OE
R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

SC3STR : x'00FD9B'

Serial 3 Status Register

8-bit access register (16-bit access is possible from even address)

SC3STR reads the status for serial interface 3.

7 Transmission Busy Flag
0: Ready to transmit
1: Transmission in progress

6 Reception Busy Flag
0: Ready to receive
1: Reception in progress

5 I²C Stop Sequence Detect
0: Undetected
1: Detected

This bit is cleared by the read or write operation of the SC3TRB register. Set 1 to the SC3REN bit.

4 Received Data
0: No received data
1: Received data

3 I²C Start Sequence Detect
0: Undetected
1: Detected

This bit is cleared by the read or write operation of the SC3TRB register. Set 1 to the SC3REN bit.

2 Framing Error
0: No error
1: Error

A framing error occurs when the stop bit is 0. Framing error data is updated whenever the stop bit is received.

1 Parity Error
0: No error
1: Error

A parity error occurs when the parity bit is 1 although it is set to 0, when the parity bit is 0 although it is set to 1, when the parity bit is odd although it is set to even, and when the parity bit is even although it is set to odd. Parity error data is updated whenever the parity bit is received.

0 Overrun Error
0: No error
1: Error

An overrun error occurs when the next data is received completely before the CPU reads the received data (SC3TRB). Overrun error data is updated whenever the last data bit (seventh or eighth bit) is received.

Do not poll the SC3RBY flag to verify the reception end in clock synchronous mode. Generate a serial 3 reception end interrupt or poll the SC3RXA flag to verify the reception end.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
SC4 TEN	SC4 REN	SC4 BRE	SC4 IIC	SC4 PTL	-	SC4 OD	SC4 ICM	SC4 LN	SC4 PTY2	SC4 PTY1	SC4 PTY0	SC4 SB	-	SC4 S1	SC4 S0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0	0/1	0/1

SC4CTR : x'00FDA0'

Serial 4 Control Register

16-bit access register

SC4CTR sets serial 4 operating conditions.

15 Transmit Enable — 0: Disable 1: Enable

14 Receive Enable — 0: Disable 1: Enable

13 Break Transmission — 0: Don't break 1: Break

12 I²C Start or Stop Sequence
0: Stop sequence output when changing this bit from 1 to 0.
1: Start sequence output when changing this bit from 0 to 1.

Change when transmission or reception is not in progress.

11 Protocol Selection
0: Asynchronous mode
1: Clock synchronous mode, I²C mode

9 Bit Order Selection
0: LSB first
1: MSB first (select only when the character length is 8-bit.)

8 I²C mode Selection — 0: I²C mode off 1: I²C mode on

7 Character Length — 0: 7-bit 1: 8-bit

6:4 Parity Bit Selection
000: None
100: 0 (output low)
101: 1 (output high)
110: Even (1s are even)
111: Odd (1s are odd)
Others: Reserved

3 Stop Bit Selection (asynchronous mode)
0: 1-bit 1: 2-bit

SBO4 Output Hold Time (clock synchronous mode)
0: More than BOSC cycles
1: More than timer 1 underflow cycles

The SBO4 output hold time is extended only when SBT4 pin is selected as serial 4 clock source.

1:0 Serial 4 Clock Source Selection
Asynchronous mode, I²C mode
01: Timer 1 underflow (1/8)
11: Timer 5 underflow (1/8)
Clock synchronous mode
00: SBT4 pin
01: Timer 1 underflow (1/8)
10: Timer 5 underflow (1/2)
11: Timer 5 underflow (1/8)

SC4TRB : x'00FDA2'

Serial 4 Transmit/ Receive Buffer

8-bit access register

SC4TRB writes the serial 4 transmit data and reads the serial 4 receive data.

Transmission starts by writing the data into this register. The data is received by reading this register. In 7-bit transfer, the MSB (bit 7) becomes 0. The data is read when an interrupt occurs or the SC4RXA flag of the SC4STR register is 1.

7	6	5	4	3	2	1	0
SC4 TRB7	SC4 TRB6	SC4 TRB5	SC4 TRB4	SC4 TRB3	SC4 TRB2	SC4 TRB1	SC4 TRB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 Serial Transmit/Receive Data

S

7	6	5	4	3	2	1	0
SC4 TBY	SC4 RBY	SC4 ISP	SC4 RXA	SC4 IST	SC4 FE	SC4 PE	SC4 OE
R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

SC4STR : x'00FDA3'

Serial 4 Status Register

8-bit access register
(16-bit access is possible from even address)

SC4STR reads the status for serial interface 4.

7 Transmission Busy Flag

0: Ready to transmit
1: Transmission in progress

6 Reception Busy Flag

0: Ready to receive
1: Reception in progress

5 I²C Stop Sequence Detect

0: Undetected
1: Detected

This bit is cleared by the read or write operation of the SC4TRB register. Set 1 to the SC4REN bit.

4 Received Data

0: No received data
1: Received data

3 I²C Start Sequence Detect

0: Undetected
1: Detected

This bit is cleared by the read or write operation of the SC4TRB register. Set 1 to the SC4REN bit.

2 Framing Error

0: No error
1: Error

A framing error occurs when the stop bit is 0. Framing error data is updated whenever the stop bit is received.

1 Parity Error

0: No error
1: Error

A parity error occurs when the parity bit is 1 although it is set to 0, when the parity bit is 0 although it is set to 1, when the parity bit is odd although it is set to even, and when the parity bit is even although it is set to odd. Parity error data is updated whenever the parity bit is received.

0 Overrun Error

0: No error
1: Error

An overrun error occurs when the next data is received completely before the CPU reads the received data (SC4TRB). Overrun error data is updated whenever the last data bit (seventh or eighth bit) is received.

Do not poll the SC4RBY flag to verify the reception end in clock synchronous mode. Generate a serial 4 reception end interrupt or poll the SC4RXA flag to verify the reception end.

TM0BC : x'00FE00'

Timer 0 Binary Counter

8-bit access register

TM0BC operates timer 0 counting.

7	6	5	4	3	2	1	0
TM0 BC7	TM0 BC6	TM0 BC5	TM0 BC4	TM0 BC3	TM0 BC2	TM0 BC1	TM0 BC0
R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM1BC : x'00FE01'

Timer 1 Binary Counter

8-bit access register (16-bit access is possible from even address)

TM1BC operates timer 1 counting.

7	6	5	4	3	2	1	0
TM1 BC7	TM1 BC6	TM1 BC5	TM1 BC4	TM1 BC3	TM1 BC2	TM1 BC1	TM1 BC0
R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM2BC : x'00FE02'

Timer 2 Binary Counter

8-bit access register

TM2BC operates timer 2 counting.

7	6	5	4	3	2	1	0
TM2 BC7	TM2 BC6	TM2 BC5	TM2 BC4	TM2 BC3	TM2 BC2	TM2 BC1	TM2 BC0
R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM3BC : x'00FE03'

Timer 3 Binary Counter

8-bit access register (16-bit access is possible from even address)

TM3BC operates timer 3 counting.

7	6	5	4	3	2	1	0
TM3 BC7	TM3 BC6	TM3 BC5	TM3 BC4	TM3 BC3	TM3 BC2	TM3 BC1	TM3 BC0
R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM4BC : x'00FE04'

Timer 4 Binary Counter

8-bit access register

TM4BC operates timer 4 counting.

7	6	5	4	3	2	1	0
TM4 BC7	TM4 BC6	TM4 BC5	TM4 BC4	TM4 BC3	TM4 BC2	TM4 BC1	TM4 BC0
R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM5BC : x'00FE05'

Timer 5 Binary Counter

8-bit access register
(16-bit access is possible from even address)

TM5BC operates timer 5 counting.

7	6	5	4	3	2	1	0
TM5 BC7	TM5 BC6	TM5 BC5	TM5 BC4	TM5 BC3	TM5 BC2	TM5 BC1	TM5 BC0
R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM6BC : x'00FE06'

Timer 6 Binary Counter

8-bit access register

TM6BC operates timer 6 counting.

7	6	5	4	3	2	1	0
TM6 BC7	TM6 BC6	TM6 BC5	TM6 BC4	TM6 BC3	TM6 BC2	TM6 BC1	TM6 BC0
R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM7BC : x'00FE07'

Timer 7 Binary Counter

8-bit access register
(16-bit access is possible from even address)

TM7BC operates timer 7 counting.

7	6	5	4	3	2	1	0
TM7 BC7	TM7 BC6	TM7 BC5	TM7 BC4	TM7 BC3	TM7 BC2	TM7 BC1	TM7 BC0
R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 BC15	TM8 BC14	TM8 BC13	TM8 BC12	TM8 BC11	TM8 BC10	TM8 BC9	TM8 BC8	TM8 BC7	TM8 BC6	TM8 BC5	TM8 BC4	TM8 BC3	TM8 BC2	TM8 BC1	TM8 BC0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM8BC : x'00FE82'

Timer 8 Binary Counter

16-bit access register

TM8BC operates timer 8 counting.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM9 BC15	TM9 BC14	TM9 BC13	TM9 BC12	TM9 BC11	TM9 BC10	TM9 BC9	TM9 BC8	TM9 BC7	TM9 BC6	TM9 BC5	TM9 BC4	TM9 BC3	TM9 BC2	TM9 BC1	TM9 BC0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM9BC : x'00FE92'

Timer 9 Binary Counter

16-bit access register

TM9BC operates timer 9 counting.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM10 BC15	TM10 BC14	TM10 BC13	TM10 BC12	TM10 BC11	TM10 BC10	TM10 BC9	TM10 BC8	TM10 BC7	TM10 BC6	TM10 BC5	TM10 BC4	TM10 BC3	TM10 BC2	TM10 BC1	TM10 BC0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM10BC : x'00FEA2'

Timer 10 Binary Counter

16-bit access register

TM10BC operates timer 10 counting.

T

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM11 BC15	TM11 BC14	TM11 BC13	TM11 BC12	TM11 BC11	TM11 BC10	TM11 BC9	TM11 BC8	TM11 BC7	TM11 BC6	TM11 BC5	TM11 BC4	TM11 BC3	TM11 BC2	TM11 BC1	TM11 BC0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM11BC : x'00FEB2'

Timer 11 Binary Counter

16-bit access register

TM11BC operates timer 11 counting.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM12 BC15	TM12 BC14	TM12 BC13	TM12 BC12	TM12 BC11	TM12 BC10	TM12 BC9	TM12 BC8	TM12 BC7	TM12 BC6	TM12 BC5	TM12 BC4	TM12 BC3	TM12 BC2	TM12 BC1	TM12 BC0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM12BC : x'00FEC2'

Timer 12 Binary Counter

16-bit access register

TM12BC operates timer 12 counting.

TM13BC : x'00FE08'

Timer 8 Binary Counter

8-bit access register

TM13BC operates timer 8 counting.

7	6	5	4	3	2	1	0
TM13 BC7	TM13 BC6	TM13 BC5	TM13 BC4	TM13 BC3	TM13 BC2	TM13 BC1	TM13 BC0
R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM14BC : x'00FE09'

Timer 14 Binary Counter

8-bit access register
(16-bit access is possible from even address)

TM14BC operates timer 14 counting.

7	6	5	4	3	2	1	0
TM14 BC7	TM14 BC6	TM14 BC5	TM14 BC4	TM14 BC3	TM14 BC2	TM14 BC1	TM14 BC0
R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM15BC : x'00FED2'

Timer 15 Binary Counter

16-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM15 BC15	TM15 BC14	TM15 BC13	TM15 BC12	TM15 BC11	TM15 BC10	TM15 BC9	TM15 BC8	TM15 BC7	TM15 BC6	TM15 BC5	TM15 BC4	TM15 BC3	TM15 BC2	TM15 BC1	TM15 BC0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM15BC operates timer 15 counting.

TM15BC is cleared on the rising of TM15IA pin.

TM0BR : x'00FE10'

Timer 0 Base Register

8-bit access register

7	6	5	4	3	2	1	0
TM0 BR7	TM0 BR6	TM0 BR5	TM0 BR4	TM0 BR3	TM0 BR2	TM0 BR1	TM0 BR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM0BR sets the timer 0 counting cycle.

TM0BR sets the counting cycle (1 to 256). The timer 0 binary counter counts the cycle of the TM0BR value +1. When BOSC is selected as the clock source, the valid range for TM0BR is 1 to 255. Otherwise, the valid range for TM0BR is 0 to 255.

TM1BR : x'00FE11'

Timer 1 Base Register

8-bit access register (16-bit access is possible from even address)

7	6	5	4	3	2	1	0
TM1 BR7	TM1 BR6	TM1 BR5	TM1 BR4	TM1 BR3	TM1 BR2	TM1 BR1	TM1 BR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM1BR sets the timer 1 counting cycle.

TM1BR sets the counting cycle (1 to 256). The timer 1 binary counter counts the cycle of the TM1BR value +1. The valid range for TM1BR is 0 to 255.

TM2BR : x'00FE12'

Timer 2 Base Register

8-bit access register

7	6	5	4	3	2	1	0
TM2 BR7	TM2 BR6	TM2 BR5	TM2 BR4	TM2 BR3	TM2 BR2	TM2 BR1	TM2 BR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM2BR sets the timer 2 counting cycle.

TM2BR sets the counting cycle (1 to 256). The timer 2 binary counter counts the cycle of the TM2BR value +1. The valid range for TM2BR is 0 to 255.

TM3BR : x'00FE13'

Timer 3 Base Register

8-bit access register
(16-bit access is possible from even address)

7	6	5	4	3	2	1	0
TM3 BR7	TM3 BR6	TM3 BR5	TM3 BR4	TM3 BR3	TM3 BR2	TM3 BR1	TM3 BR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM3BR sets the timer 3 counting cycle.

TM3BR sets the counting cycle (1 to 256). The timer 3 binary counter counts the cycle of the TM3BR value +1. The valid range for TM3BR is 0 to 255.

7	6	5	4	3	2	1	0
TM4 BR7	TM4 BR6	TM4 BR5	TM4 BR4	TM4 BR3	TM4 BR2	TM4 BR1	TM4 BR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM4BR : x'00FE14'

Timer 4 Base Register

8-bit access register

TM4BR sets the timer 4 counting cycle.

TM4BR sets the counting cycle (1 to 256). The timer 4 binary counter counts the cycle of the TM4BR value +1. The valid range for TM4BR is 0 to 255.

7	6	5	4	3	2	1	0
TM5 BR7	TM5 BR6	TM5 BR5	TM5 BR4	TM5 BR3	TM5 BR2	TM5 BR1	TM5 BR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM5BR : x'00FE15'

Timer 5 Base Register

8-bit access register
(16-bit access is possible from even address)

TM5BR sets the timer 5 counting cycle.

TM5BR sets the counting cycle (1 to 256). The timer 5 binary counter counts the cycle of the TM5BR value +1. When BOSC is selected as the clock source, the valid range for TM5BR is 1 to 255. Otherwise, the valid range for TM5BR is 0 to 255.

TM6BR : x'00FE16'

Timer 6 Base Register

8-bit access register

7	6	5	4	3	2	1	0
TM6 BR7	TM6 BR6	TM6 BR5	TM6 BR4	TM6 BR3	TM6 BR2	TM6 BR1	TM6 BR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM6BR sets the timer 6 counting cycle.

TM6BR sets the counting cycle (1 to 256). The timer 6 binary counter counts the cycle of the TM6BR value +1. The valid range for TM6BR is 0 to 255.

TM7BR : x'00FE17'

Timer 7 Base Register

8-bit access register
(16-bit access is possible from even address)

7	6	5	4	3	2	1	0
TM7 BR7	TM7 BR6	TM7 BR5	TM7 BR4	TM7 BR3	TM7 BR2	TM7 BR1	TM7 BR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM7BR sets the timer 7 counting cycle.

TM7BR sets the counting cycle (1 to 256). The timer 7 binary counter counts the cycle of the TM7BR value +1. The valid range for TM7BR is 0 to 255.

TM13BR : x'00FE18'

Timer 13 Base Register

8-bit access register

7	6	5	4	3	2	1	0
TM13 BR7	TM13 BR6	TM13 BR5	TM13 BR4	TM13 BR3	TM13 BR2	TM13 BR1	TM13 BR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

Sets the timer 13 counting cycle.

Sets the counting cycle (1 to 256). The timer 13 binary counter counts the cycle of the TM13BR value +1. The valid range for TM13BR is 0 to 255.

TM14BR : x'00FE19'

Timer 14 Base Register

8-bit access register
(16-bit access is possible from even address)

7	6	5	4	3	2	1	0
TM14 BR7	TM14 BR6	TM14 BR5	TM14 BR4	TM4 BR3	TM14 BR2	TM14 BR1	TM14 BR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

Sets the timer 14 counting cycle.

Sets the counting cycle (1 to 256). The timer 14 binary counter counts the cycle of the TM14BR value +1. The valid range for TM14BR is 0 to 255.

TM0MD : x'00FE20'

Timer 0 Mode Register

8-bit access register

7	6	5	4	3	2	1	0
TM0 EN	TM0 LD	-	-	-	-	TM0 S1	TM0 S0
R/W	R/W	R	R	R	R	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0	0	0	0	0/1	0/1

TM0MD sets the timer 0 operating conditions.

7 **TM0BC Count**
0: Disable
1: Enable

6 **Load TM0BR to TM0BC**
0: Disable
1: Enable
Reset the 1/2 divisor circuit.

1:0 **Clock Source Selection**
00: BOSC/2
01: XI/4
10: BOSC
11: TM0IO pin input

When BOSC is selected as the clock source, the valid range for TM0BR is 1 to 255.

TM1MD : x'00FE21'

Timer 1 Mode Register

8-bit access register
(16-bit access is possible from even address)

7	6	5	4	3	2	1	0
TM1 EN	TM1 LD	-	-	-	-	TM1 S1	TM1 S0
R/W	R/W	R	R	R	R	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0	0	0	0	0/1	0/1

TM1MD sets the timer 1 operating conditions.

7 **TM1BC Count**
0: Disable
1: Enable

6 **Load TM1BR to TM1BC**
0: Disable
1: Enable
Reset the 1/2 divisor circuit.

1:0 **Clock Source Selection**
00: BOSC/2
01: Timer 0 underflow
10: Timer 0 cascade
11: Timer 4 underflow

TM2MD : x'00FE22'

Timer 2 Mode Register

8-bit access register

TM2MD sets the timer 2 operating conditions.

7	6	5	4	3	2	1	0
TM2 EN	TM2 LD	-	-	-	-	TM2 S1	TM2 S0
R/W	R/W	R	R	R	R	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0	0	0	0	0/1	0/1

7 TM2BC Count
0: Disable
1: Enable

6 Load TM2BR to TM2BC
0: Disable
1: Enable
Reset the 1/2 divisor circuit.

1:0 Clock Source Selection
00: BOSC/2
01: Timer 0 underflow
10: Timer 1 cascade
11: Timer 4 underflow

TM3MD : x'00FE23'

Timer 3 Mode Register

8-bit access register
(16-bit access is possible from even address)

TM3MD sets the timer 3 operating conditions.

7	6	5	4	3	2	1	0
TM3 EN	TM3 LD	-	-	-	-	TM3 S1	TM3 S0
R/W	R/W	R	R	R	R	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0	0	0	0	0/1	0/1

7 TM3BC Count
0: Disable
1: Enable

6 Load TM3BR to TM3BC
0: Disable
1: Enable
Reset the 1/2 divisor circuit.

1:0 Clock Source Selection
00: BOSC/2
01: Timer 0 underflow
10: Timer 2 cascade
11: Timer 4 underflow

TM4MD : x'00FE24'

Timer 4 Mode Register

8-bit access register

TM4MD sets the timer 4 operating conditions.

7	6	5	4	3	2	1	0
TM4 EN	TM4 LD	-	-	-	-	TM4 S1	TM4 S0
R/W	R/W	R	R	R	R	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0	0	0	0	0/1	0/1

7 TM4BC Count
0: Disable
1: Enable

6 Load TM4BR to TM4BC
0: Disable
1: Enable
Reset the 1/2 divisor circuit.

1:0 Clock Source Selection
00: BOSC/2
01: Timer 0 underflow
10: Timer 3 cascade
11: TM4IO pin input

TM5MD : x'00FE25'

Timer 5 Mode Register

8-bit access register
(16-bit access is possible from even address)

TM5MD sets the timer 5 operating conditions.

7	6	5	4	3	2	1	0
TM5 EN	TM5 LD	-	-	-	-	TM5 S1	TM5 S0
R/W	R/W	R	R	R	R	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0	0	0	0	0/1	0/1

7 TM5BC Count
0: Disable
1: Enable

6 Load TM5BR to TM5BC
0: Disable
1: Enable
Reset the 1/2 divisor circuit.

1:0 Clock Source Selection
00: BOSC/2
01: Timer 0 underflow
10: Timer 4 cascade
11: BOSC

When selecting BOSC as the clock source (set '11'), the valid range for TM5BR is 1 to 255.

TM6MD : x'00FE26'

Timer 6 Mode Register

8-bit access register

TM6MD sets the timer 6 operating conditions.

7	6	5	4	3	2	1	0
TM6 EN	TM6 LD	-	-	-	-	TM6 S1	TM6 S0
R/W	R/W	R	R	R	R	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0	0	0	0	0/1	0/1

7 TM6BC Count
0: Disable
1: Enable

6 Load TM6BR to TM6BC
0: Disable
1: Enable
Reset the 1/2 divisor circuit.

1:0 Clock Source Selection
00: XI/4
01: Timer 0 underflow
10: Timer 5 cascade
11: Timer 4 underflow

TM7MD : x'00FE27'

Timer 7 Mode Register

8-bit access register
(16-bit access is possible from even address)

TM7MD sets the timer 7 operating conditions.

7	6	5	4	3	2	1	0
TM7 EN	TM7 LD	-	-	-	-	TM7 S1	TM7 S0
R/W	R/W	R	R	R	R	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0	0	0	0	0/1	0/1

7 TM7BC Count
0: Disable
1: Enable

6 Load TM7BR to TM7BC
0: Disable
1: Enable
Reset the 1/2 divisor circuit.

1:0 Clock Source Selection
00: XI/4
01: Timer 0 underflow
10: Timer 6 cascade
11: TM7IO pin input

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 EN	TM8 NLD	-	-	TM8 UD1	TM8 UD0	TM8 TGE	TM8 ONE	TM8 MD1	TM8 MD0	TM8 ECLR	TM8 LP	TM8 ASEL	TM8 S2	TM8 S1	TM8 S0
R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM8MD : x'00FE80'

Timer 8 Mode Register

16-bit access register

TM8MD sets the timer 8 operating conditions.

15 TM8BC Count
0: Disable 1: Enable

14 TM8BC, T.F.F., RS.F.F. Operation
0: Set TM8BC, T.F.F., RS.F.F. to 0
1: Operate TM8BC, T.F.F., RS.F.F.

11:10 Up/Down Counter Mode Selection (Ignored when two-phase encoding is selected.)
00: Up counter
01: Down counter
10: Up when TM8IOA pin is high, down when TM8IOA pin is low
11: Up when TM8IOB pin is high, down when TM8IOB pin is low

9 Count Start External Trigger Enable
0: Disable
1: Start counting on the falling edge of TM8IOB pin

8 Counter Operating Mode Selection
0: Repeat 1: One-shot counting

During repeat counting, hold the TM8EN flag state. During one-shot counting, set the TM8EN flag to 0 when TM8BC=TM8CA.

7:6 TM8CA, TM8CB Operating Mode Selection
00: Compare register (single buffer)
01: Compare register (double buffer)
10: Capture A when TM8IOA pin is high, Capture B when TM8IOA pin is low
11: Capture A when TM8IOA pin is high, Capture B when TM8IOB pin is high

5 TM8BC Clear
0: Don't clear 1: Clear

4 TM8BC Count Range
0: 0 to FFFF 1: 0 to TM8CA

3 TM8IOA Pin Output
0: RS.F.F. output (one-phase PWM)
1: T.F.F. output (two-phase PWM)

2:0 Clock Source Selection
000: Timer 0 underflow
001: Timer 4 underflow
010: TM8IOB pin
011: BOSC/2
100: Two-phase encoder (4x) of TM8IOA pin, TM8IOB pin
101: Two-phase encoder (1x) of TM8IOA pin, TM8IOB pin
110: TM8IC pin

7	6	5	4	3	2	1	0
-	-	-	-	-	TM8 IC	TM8 IOB	TM8 IOA
R	R	R	R	R	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0/1	0/1	0/1

TM8MD2 : x'00FE8E'

Timer 8 Mode Register 2

8-bit access register

TM8MD2 sets the timer 8 operating conditions.

2 **TM8IC Pin Input Edge Select**

0: Change TM8IOB pin output on the rising edge

1: Change TM8IOB pin output on the falling edge

1 **TM8IOB Pin Output**

0: Positive

1: Negative

0 **TM8IOA Pin Output**

0: Positive

1: Negative

T

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM9 EN	TM9 NLD	-	-	TM9 UD1	TM9 UD0	TM9 TGE	TM9 ONE	TM9 MD1	TM9 MD0	TM9 ECLR	TM9 LP	TM9 ASEL	TM9 S2	TM9 S1	TM9 S0
R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM9MD : x'00FE90'

Timer 9 Mode Register

16-bit access register

TM9MD sets the timer 9 operating conditions.

15 TM9BC Count — 0: Disable 1: Enable

14 TM9BC, T.F.F., RS.F.F. Operation
0: Set TM9BC, T.F.F., RS.F.F. to 0
1: Operate TM9BC, T.F.F., RS.F.F.

11:10 Up/Down Counter Mode Selection (Ignored when two-phase encoding is selected.)
00: Up counter
01: Down counter
10: Up when TM9IOA pin is high, down when TM9IOA pin is low
11: Up when TM9IOB pin is high, down when TM9IOB pin is low

9 Count Start External Trigger Enable
0: Disable
1: Start counting on the falling edge of TM9IOB pin

8 Counter Operating Mode Selection
0: Repeat 1: One-shot counting

During repeat counting, hold the TM9EN flag state. During one-shot counting, set the TM9EN flag to 0 when TM9BC=TM9CA.

7:6 TM9CA, TM9CB Operating Mode Selection
00: Compare register (single buffer)
01: Compare register (double buffer)
10: Capture A when TM9IOA pin is high, Capture B when TM9IOA pin is low
11: Capture A when TM9IOA pin is high, Capture B when TM9IOB pin is high

5 TM9BC Clear
0: Don't clear
1: Clear (when external synchronization is used)

4 TM9BC Count Range
0: 0 to FFFF 1: 0 to TM9CA

3 TM9IOA Pin Output
0: RS.F.F. output (one-phase PWM)
1: T.F.F. output (two-phase PWM)

2:0 Clock Source Selection
000: Timer 0 underflow
001: Timer 4 underflow
010: TM9IOB pin
011: BOSC/2
100: Two-phase encoder (4x) of TM9IOA pin, TM9IOB pin
101: Two-phase encoder (1x) of TM9IOA pin, TM9IOB pin

TM9MD2 : x'00FE9E'

Timer 9 Mode Register 2

8-bit access register

TM9MD2 sets the conditions to clear the timer 9 binary counter.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	TM9 BC	TM9 IB
R	R	R	R	R	R	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0/1	0/1

1 TM9BC Clear Using TM9IOB Pin Input

0: Don't clear
1: Clear

0 TM9IB Pin Input Polarity

0: Positive
1: Negative

T

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM10 EN	TM10 NLD	-	-	TM10 UD1	TM10 UD0	TM10 TGE	TM10 ONE	TM10 MD1	TM10 MD0	TM10 ECLR	TM10 LP	TM10 ASEL	TM10 S2	TM10 S1	TM10 S0
R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM10MD : x'00FEA0'

Timer 10 Mode Register

16-bit access register

TM10MD sets the timer 10 operating conditions.

15 TM10BC Count
0: Disable 1: Enable

14 TM10BC, T.F.F., RS.F.F. Operation
0: Set TM10BC, T.F.F., RS.F.F. to 0
1: Operate TM10BC, T.F.F., RS.F.F.

11:10 Up/Down Counter Mode Selection (Ignored when two-phase encoding is selected.)
00: Up counter
01: Down counter
10: Up when TM10IOA pin is high, down when TM10IOA pin is low
11: Up when TM10IOB pin is high, down when TM10IOB pin is low

9 Count Start External Trigger Enable
0: Disable
1: Start counting on the falling edge of TM10IOB pin

8 Counter Operating Mode Selection
0: Repeat 1: One-shot counting

During repeat counting, hold the TM10EN flag state. During one-shot counting, set the TM10EN flag to 0 when TM10BC = TM10CA.

7:6 TM10CA, TM10CB Operating Mode Selection
00: Compare register (single buffer)
01: Compare register (double buffer)
10: Capture A when TM10IOA pin is high, Capture B when TM10IOA pin is low
11: Capture A when TM10IOA pin is high, Capture B when TM10IOB pin is high

5 TM10BC Clear
0: Don't clear
1: Clear (when external synchronization is used)

4 TM10BC Count Range
0: 0 to FFFF 1: 0 to TM10CA

3 TM10IOA Pin Output
0: RS.F.F. output (one-phase PWM)
1: T.F.F. output (two-phase PWM)

2:0 Clock Source Selection
000: Timer 0 underflow
001: Timer 4 underflow
010: TM10IOB pin (Rising, falling, both edges)
011: BOSC/2
100: Two-phase encoder (4x) of TM10IOA pin, TM10IOB pin
101: Two-phase encoder (1x) of TM10IOA pin, TM10IOB pin

The TM10IOB pin edge is set in the TM10MD2 register.

TM10MD2 : x'00FEAE'

Timer 10 Mode Register 2

8-bit access register

TM10MD2 sets the TM10IB pin input edge.

7	6	5	4	3	2	1	0
-	-	-	-	-	-	TM10 IB1	TM10 IB0
R	R	R	R	R	R	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0/1	0/1

1:0 TM10IB Pin Input Edge

00: Rising edge
01: Falling edge
10: Both edges

T

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM11 EN	TM11 NLD	-	-	TM11 UD1	TM11 UD0	TM11 TGE	TM11 ONE	TM11 MD1	TM11 MD0	TM11 ECLR	TM11 LP	TM11 ASEL	TM11 S2	TM11 S1	TM11 S0
R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM11MD : x'00FEB0'

Timer 11 Mode Register

16-bit access register

TM11MD sets the timer 11 operating conditions.

15 TM11BC Count
0: Disable 1: Enable

14 TM11BC, T.F.F., RS.F.F. Operation
0: Set TM11BC, T.F.F., RS.F.F. to 0
1: Operate TM11BC, T.F.F., RS.F.F.

11:10 Up/Down Counter Mode Selection (Ignored when two-phase encoding is selected.)
00: Up counter
01: Down counter
10: Up when TM11IOA pin is high, down when TM11IOA pin is low
11: Up when TM11IOB pin is high, down when TM11IOB pin is low

9 Count Start External Trigger Enable
0: Disable
1: Start counting on the falling edge of TM11IOB pin

8 Counter Operating Mode Selection
0: Repeat 1: One-shot counting

During repeat counting, hold the TM11EN flag state. During one-shot counting, set the TM11EN flag to 0 when TM11BC = TM11CA.

7:6 TM11CA, TM11CB Operating Mode Selection
00: Compare register (single buffer)
01: Compare register (double buffer)
10: Capture A when TM11IOA pin is high, Capture B when TM11IOA pin is low
11: Capture A when TM11IOA pin is high, Capture B when TM11IOB pin is high

5 TM11BC Clear
0: Don't clear
1: Clear (when external synchronization is used)

4 TM11BC Count Range
0: 0 to FFFF 1: 0 to TM11CA

3 TM11IOA Pin Output
0: RS.F.F. output (one-phase PWM)
1: T.F.F. output (two-phase PWM)

2:0 Clock Source Selection
000: Timer 0 underflow
001: Timer 4 underflow
010: TM11IOB pin
011: BOSC/2
100: Two-phase encoder (4x) of TM11IOA pin, TM11IOB pin
101: Two-phase encoder (1x) of TM11IOA pin, TM11IOB pin

The TM10IOB pin edge is set in the TM10MD2 register.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM12 EN	TM12 NLD	-	-	TM12 UD1	TM12 UD0	TM12 TGE	TM12 ONE	TM12 MD1	TM12 MD0	TM12 ECLR	TM12 LP	TM12 ASEL	TM12 S2	TM12 S1	TM12 S0
R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM12MD : x'00FEC0'

Timer 12 Mode Register

16-bit access register

TM12MD sets the timer 12 operating conditions.

15 TM12BC Count
0: Disable 1: Enable

14 TM12BC, T.F.F., RS.F.F. Operation
0: Set TM12BC, T.F.F., RS.F.F. to 0
1: Operate TM12BC, T.F.F., RS.F.F.

11:10 Up/Down Counter Mode Selection (Ignored when two-phase encoding is selected.)
00: Up counter
01: Down counter
10: Up when TM12IOA pin is high, down when TM12IOA pin is low
11: Up when TM12IOB pin is high, down when TM12IOB pin is low

9 Count Start External Trigger Enable
0: Disable
1: Start counting on the falling edge of TM12IOB pin

8 Counter Operating Mode Selection
0: Repeat 1: One-shot counting

During repeat counting, hold the TM12EN flag state. During one-shot counting, set the TM12EN flag to 0 when TM12BC = TM12CA.

7:6 TM12CA, TM12CB Operating Mode Selection
00: Compare register (single buffer)
01: Compare register (double buffer)
10: Capture A when TM12IOA pin is high, Capture B when TM12IOA pin is low
11: Capture A when TM12IOA pin is high, Capture B when TM12IOB pin is high

5 TM12BC Clear
0: Don't clear
1: Clear (when external synchronization is used)

4 TM12BC Count Range
0: 0 to FFFF 1: 0 to TM12CA

3 TM12IOA Pin Output
0: RS.F.F. output (one-phase PWM)
1: T.F.F. output (two-phase PWM)

2:0 Clock Source Selection
000: Timer 0 underflow
001: Timer 4 underflow
010: TM12IOB pin
011: BOSC/2
100: Two-phase encoder (4x) of TM12IOA pin, TM12IOB pin
101: Two-phase encoder (1x) of TM12IOA pin, TM12IOB pin

7	6	5	4	3	2	1	0
TM13 EN	TM13 LD	TM13 CLR	-	-	TM13 OB	TM13 OA	TM13 S
R/W	R/W	R/W	R	R	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0	0	0/1	0/1	0/1

TM13MD : x'00FE28'

Timer 13 Mode Register

8-bit access register

TM13MD sets the timer 13 operating conditions.

7	**TM13BC Count**	0: Disable	1: Enable
6	**Load TM13BR to TM13BC**	0: Disable	1: Enable
5	**TM13OA, TM13OB Signal Level**	0: Disable 1: Enable *	
2	**TM13OB Output Edge Select**	0: Positive logic	1: Negative logic
1	**TM13OA Output Edge Select**	0: Positive logic	1: Negative logic
0	**Clock Source Selection**	0: BOSC/2 1: Timer 0 underflow	

* Whenever the timer 13 counting is stopped, TM13OA and TM13OB signals go low when the positive logic is selected as output edge, while these signals go high when the negative logic is selected as output edge.

7	6	5	4	3	2	1	0
TM14 EN	TM14 LD	TM14 CLR	-	-	TM14 OB	TM14 OA	TM14 S
R/W	R/W	R	R	R	R	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0	0	0	0	0/1	0/1

TM14MD : x'00FE29'

Timer 14 Mode Register

8-bit access register

TM14MD sets the timer 14 operating conditions.

7	**TM14BC Count**	0: Disable	1: Enable
6	**Load TM14BR to TM14BC**	0: Disable	1: Enable
5	**TM14OA, TM14OB Signal Level**	0: Disable 1: Enable *	
2	**TM14OB Output Edge Select**	0: Positive logic	1: Negative logic
1	**TM14OA Output Edge Select**	0: Positive logic	1: Negative logic
0	**Clock Source Selection**	0: BOSC/2 1: Timer 0 underflow	

* Whenever the timer 14 counting is stopped, TM14OA and TM14OB signals go low when the positive logic is selected as output edge, while these signals go high when the negative logic is selected as output edge.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM15 BC	reserv ed	-	-	-	-	-	-	-	-	-	-	-	-	TM15 CLK1	TM15 CLK0
R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0	0	0	0	0	0	0	0	0	0	0	0	0/1	0/1

TM15MD : x'00FED0'

Timer 15 Mode Register

16-bit access register

TM15MD sets the timer 15 operating conditions.

15 **TM15BC Count** 0: Disable 1: Enable

14 **Reserved** Set to 0

1:0 **Clock Source Selection**
00: Timer 0 underflow
01: IRQ4 pin
10: BOSC/2
11: BOSC

T

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 CA15	TM8 CA14	TM8 CA13	TM8 CA12	TM8 CA11	TM8 CA10	TM8 CA9	TM8 CA8	TM8 CA7	TM8 CA6	TM8 CA5	TM8 CA4	TM8 CA3	TM8 CA2	TM8 CA1	TM8 CA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM8CA : x'00FE84'

Timer 8 Compare/ Capture Register A

16-bit access register

TM8CA sets the timer 8 counting cycle.

The timer 8 binary counter counts the cycle of the TM8CA value +1. TM8CA changes PWM and generates a timer 8 capture A interrupt. When capture is selected, this register is read only. A timer 8 capture A interrupt is generated when capture occurs. When compare is selected, set the PWM cycle. When this register matches the timer 8 binary counter, a timer 8 capture A interrupt occurs.
This register write only 16-bit data. Use the MOV instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T8C AX15	T8C AX14	T8C AX13	T8C AX12	T8C AX11	T8C AX10	T8C AX9	T8C AX8	T8C AX7	T8C AX6	T8C AX5	T8C AX4	T8C AX3	T8C AX2	T8C AX1	T8C AX0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

TM8CAX : x'00FE86'

Timer 8 Compare/ Capture Register Set A

16-bit access register

This register is valid only when the associated compare register is set to the double-buffer mode. The TM8CAX cannot read or write. The contents of TM8CA are loaded to TM8CAX by write signal.

TM8CAX sets the PWM cycle. When TM8BC=TM8CAX, a timer 8 capture A interrupt occurs. The contents of TM8CA are loaded to TM8CAX by a timer 8 capture A interrupt and TM8CAX prevents the PWM loss.

This register writes only 16-bit data. Use the MOV instruction to set the data.

TM8CB : x'00FE88'

Timer 8 Compare/ Capture Register B
16-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM8 CB15	TM8 CB14	TM8 CB13	TM8 CB12	TM8 CB11	TM8 CB10	TM8 CB9	TM8 CB8	TM8 CB7	TM8 CB6	TM8 CB5	TM8 CB4	TM8 CB3	TM8 CB2	TM8 CB1	TM8 CB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM8CB sets the timer 8 PWM duty, changes PWM and generates a timer 8 capture B interrupt.

When capture is selected, this register is read only. A timer 8 capture B interrupt is generated when capture occurs. When compare is selected, set the PWM duty. When this register matches the timer 8 binary counter, a timer 8 capture B interrupt occurs.

This register write only 16-bit data. Use the MOV instruction to set the data.

TM8CBX : x'00FE8A'

Timer 8 Compare/ Capture Register Set B
16-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T8C BX15	T8C BX14	T8C BX13	T8C BX12	T8C BX11	T8C BX10	T8C BX9	T8C BX8	T8C BX7	T8C BX6	T8C BX5	T8C BX4	T8C BX3	T8C BX2	T8C BX1	T8C BX0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

This register is valid only when the associated compare register is set to the double-buffer mode. TM8CBX cannot read or write. The contents of TM8CB are loaded to TM8CBX by write signal.

TM8CBX sets the PWM cycle. When TM8BC=TM8CBX, a timer 8 capture B interrupt occurs. The contents of TM8CB are loaded to TM8CBX by a timer 8 capture B interrupt and TM8CBX prevents the PWM loss.

This register writes only 16-bit data. Use the MOV instruction to set the data.

TM9CA : x'00FE94'

Timer 9 Compare/ Capture Register A
16-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM9 CA15	TM9 CA14	TM9 CA13	TM9 CA12	TM9 CA11	TM9 CA10	TM9 CA9	TM9 CA8	TM9 CA7	TM9 CA6	TM9 CA5	TM9 CA4	TM9 CA3	TM9 CA2	TM9 CA1	TM9 CA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM9CA sets the timer 9 counting cycle.

The timer 9 binary counter counts the cycle of the TM9CA value +1. TM9CA changes PWM and generates a timer 9 capture A interrupt. When capture is selected, this register is read only. A timer 9 capture A interrupt is generated when capture occurs. When compare is selected, set the PWM cycle. When this register matches the timer 9 binary counter, a timer 9 capture A interrupt occurs.
This register write only 16-bit data. Use the MOV instruction to set the data.

TM9CAX : x'00FE96'

Timer 9 Compare/ Capture Register Set A
16-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T9C AX15	T9C AX14	T9C AX13	T9C AX12	T9C AX11	T9C AX10	T9C AX9	T9C AX8	T9C AX7	T9C AX6	T9C AX5	T9C AX4	T9C AX3	T9C AX2	T9C AX1	T9C AX0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

This register is valid only when the associated compare register is set to the double-buffer mode. The TM9CAX cannot read or write. The contents of TM9CA are loaded to TM9CAX by write signal.

TM9CAX sets the PWM cycle. When TM9BC=TM9CAX, a timer 9 capture A interrupt occurs. The contents of TM9CA are loaded to TM9CAX by a timer 9 capture A interrupt and TM9CAX prevents the PWM loss.

This register writes only 16-bit data. Use the MOV instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM9 CB15	TM9 CB14	TM9 CB13	TM9 CB12	TM9 CB11	TM9 CB10	TM9 CB9	TM9 CB8	TM9 CB7	TM9 CB6	TM9 CB5	TM9 CB4	TM9 CB3	TM9 CB2	TM9 CB1	TM9 CB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM9CB : x'00FE98'

Timer 9 Compare/ Capture Register B

16-bit access register

TM9CB sets the timer 9 PWM duty, changes PWM and generates a timer 9 capture B interrupt.

When capture is selected, this register is read only. A timer 9 capture B interrupt is generated when capture occurs. When compare is selected, set the PWM duty. When this register matches the timer 9 binary counter, a timer 9 capture B interrupt occurs.

This register write only 16-bit data. Use the MOV instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T9C BX15	T9C BX14	T9C BX13	T9C BX12	T9C BX11	T9C BX10	T9C BX9	T9C BX8	T9C BX7	T9C BX6	T9C BX5	T9C BX4	T9C BX3	T9C BX2	T9C BX1	T9C BX0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

TM9CBX : x'00FE9A'

Timer 9 Compare/ Capture Register Set B

16-bit access register

This register is valid only when the associated compare register is set to the double-buffer mode. TM9CBX cannot read or write. The contents of TM9CB are loaded to TM9CBX by write signal.

TM9CBX sets the PWM cycle. When TM9BC=TM9CBX, a timer 9 capture B interrupt occurs. The contents of TM9CB are loaded to TM9CBX by a timer 9 capture B interrupt and TM9CBX prevents the PWM loss.

This register writes only 16-bit data. Use the MOV instruction to set the data.

TM10CA : x'00FEA4'

Timer 10 Compare/ Capture Register A

16-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM10 CA15	TM10 CA14	TM10 CA13	TM10 CA12	TM10 CA11	TM10 CA10	TM10 CA9	TM10 CA8	TM10 CA7	TM10 CA6	TM10 CA5	TM10 CA4	TM10 CA3	TM10 CA2	TM10 CA1	TM10 CA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM10CA sets the timer 10 counting cycle.

The timer 10 binary counter counts the cycle of the TM10CA value +1. TM10CA changes PWM and generates a timer 10 capture A interrupt.
When capture is selected, this register is read only. A timer 10 capture A interrupt is generated when capture occurs. When compare is selected, set the PWM cycle. When this register matches the timer 10 binary counter, a timer 10 capture A interrupt occurs.
This register write only 16-bit data. Use the MOV instruction to set the data.

TM10CAX : x'00FEA6'

Timer 10 Compare/ Capture Register Set A

16-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T10C AX15	T10C AX14	T10C AX13	T10C AX12	T10C AX11	T10C AX10	T10C AX9	T10C AX8	T10C AX7	T10C AX6	T10C AX5	T10C AX4	T10C AX3	T10C AX2	T10C AX1	T10C AX0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

This register is valid only when the associated compare register is set to the double-buffer mode. The TM10CAX cannot read or write. The contents of TM10CA are loaded to TM10CAX by write signal.

TM10CAX sets the PWM cycle. When TM10BC=TM10CAX, a timer 10 capture A interrupt occurs. The contents of TM10CA are loaded to TM10CAX by a timer 10 capture A interrupt and TM10CAX prevents the PWM loss.

This register writes only 16-bit data. Use the MOV instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM10 CB15	TM10 CB14	TM10 CB13	TM10 CB12	TM10 CB11	TM10 CB10	TM10 CB9	TM10 CB8	TM10 CB7	TM10 CB6	TM10 CB5	TM10 CB4	TM10 CB3	TM10 CB2	TM10 CB1	TM10 CB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM10CB : x'00FEA8'

Timer 10 Compare/ Capture Register B
16-bit access register

TM10CB sets the timer10 PWM duty, changes PWM and generates a timer 10 capture B interrupt.

When capture is selected, this register is read only. A timer 10 capture B interrupt is generated when capture occurs. When compare is selected, set the PWM duty. When this register matches the timer 10 binary counter, a timer 10 capture B interrupt occurs.

This register write only 16-bit data. Use the MOV instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T10C BX15	T10C BX14	T10C BX13	T10C BX12	T10C BX11	T10C BX10	T10C BX9	T10C BX8	T10C BX7	T10C BX6	T10C BX5	T10C BX4	T10C BX3	T10C BX2	T10C BX1	T10C BX0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

TM10CBX : x'00FEAA'

Timer 10 Compare/ Capture Register Set B
16-bit access register

This register is valid only when the associated compare register is set to the double-buffer mode. TM10CBX cannot read or write. The contents of TM10CB are loaded to TM10CBX by write signal.

TM10CBX sets the PWM cycle. When TM10BC=TM10CBX, a timer 10 capture B interrupt occurs. The contents of TM10CB are loaded to TM10CBX by a timer 10 capture B interrupt and TM10CBX prevents the PWM loss.

This register writes only 16-bit data. Use the MOV instruction to set the data.

TM11CA : x'00FEB4'

Timer 11 Compare/ Capture Register A

16-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM11 CA15	TM11 CA14	TM11 CA13	TM11 CA12	TM11 CA11	TM11 CA10	TM11 CA9	TM11 CA8	TM11 CA7	TM11 CA6	TM11 CA5	TM11 CA4	TM11 CA3	TM11 CA2	TM11 CA1	TM11 CA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM11CA sets the timer 11 counting cycle.

The timer 11 binary counter counts the cycle of the TM11CA value +1. TM11CA changes PWM and generates a timer 11 capture A interrupt.
When capture is selected, this register is read only. A timer 11 capture A interrupt is generated when capture occurs. When compare is selected, set the PWM cycle. When this register matches the timer 11 binary counter, a timer 11 capture A interrupt occurs.
This register write only 16-bit data. Use the MOV instruction to set the data.

TM11CAX : x'00FEB6'

Timer 11 Compare/ Capture Register Set A

16-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T11C AX15	T11C AX14	T11C AX13	T11C AX12	T11C AX11	T11C AX10	T11C AX9	T11C AX8	T11C AX7	T11C AX6	T11C AX5	T11C AX4	T11C AX3	T11C AX2	T11C AX1	T11C AX0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

This register is valid only when the associated compare register is set to the double-buffer mode. The TM11CAX cannot read or write. The contents of TM11CA are loaded to TM11CAX by write signal.

TM11CAX sets the PWM cycle. When TM11BC=TM11CAX, a timer 11 capture A interrupt occurs. The contents of TM11CA are loaded to TM11CAX by a timer 11 capture A interrupt and TM11CAX prevents the PWM loss.

This register writes only 16-bit data. Use the MOV instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM11 CB15	TM11 CB14	TM11 CB13	TM11 CB12	TM11 CB11	TM11 CB10	TM11 CB9	TM11 CB8	TM11 CB7	TM11 CB6	TM11 CB5	TM11 CB4	TM11 CB3	TM11 CB2	TM11 CB1	TM11 CB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM11CB : x'00FEB8'

Timer 11 Compare/ Capture Register B

16-bit access register

TM11CB sets the timer11 PWM duty, changes PWM and generates a timer 11 capture B interrupt.

When capture is selected, this register is read only. A timer 11 capture B interrupt is generated when capture occurs. When compare is selected, set the PWM duty. When this register matches the timer 11 binary counter, a timer 11 capture B interrupt occurs.

This register write only 16-bit data. Use the MOV instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T11C BX15	T11C BX14	T11C BX13	T11C BX12	T11C BX11	T11C BX10	T11C BX9	T11C BX8	T11C BX7	T11C BX6	T11C BX5	T11C BX4	T11C BX3	T11C BX2	T11C BX1	T11C BX0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

TM11CBX : x'00FEBA'

Timer 11 Compare/ Capture Register Set B

16-bit access register

This register is valid only when the associated compare register is set to the double-buffer mode. TM11CBX cannot read or write. The contents of TM11CB are loaded to TM11CBX by write signal.

TM11CBX sets the PWM cycle. When TM11BC=TM11CBX, a timer 11 capture B interrupt occurs. The contents of TM11CB are loaded to TM11CBX by a timer 11 capture B interrupt and TM11CBX prevents the PWM loss.

This register writes only 16-bit data. Use the MOV instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM12 CA15	TM12 CA14	TM12 CA13	TM12 CA12	TM12 CA11	TM12 CA10	TM12 CA9	TM12 CA8	TM12 CA7	TM12 CA6	TM12 CA5	TM12 CA4	TM12 CA3	TM12 CA2	TM12 CA1	TM12 CA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM12CA : x'00FEC4'

Timer 12 Compare/ Capture Register A

16-bit access register

TM12CA sets the timer 12 counting cycle.

The timer 12 binary counter counts the cycle of the TM12CA value +1. TM12CA changes PWM and generates a timer 12 capture A interrupt.
When capture is selected, this register is read only. A timer 12 capture A interrupt is generated when capture occurs. When compare is selected, set the PWM cycle. When this register matches the timer 12 binary counter, a timer 12 capture A interrupt occurs.
This register write only 16-bit data. Use the MOV instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T12C AX15	T12C AX14	T12C AX13	T12C AX12	T12C AX11	T12C AX10	T12C AX9	T12C AX8	T12C AX7	T12C AX6	T12C AX5	T12C AX4	T12C AX3	T12C AX2	T12C AX1	T12C AX0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

TM12CAX : x'00FEC6'

Timer 12 Compare/ Capture Register Set A

16-bit access register

This register is valid only when the associated compare register is set to the double-buffer mode. The TM12CAX cannot read or write. The contents of TM12CA are loaded to TM12CAX by write signal.

TM12CAX sets the PWM cycle. When TM12BC=TM12CAX, a timer 12 capture A interrupt occurs. The contents of TM12CA are loaded to TM12CAX by a timer 12 capture A interrupt and TM12CAX prevents the PWM loss.

This register writes only 16-bit data. Use the MOV instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM12 CB15	TM12 CB14	TM12 CB13	TM12 CB12	TM12 CB11	TM12 CB10	TM12 CB9	TM12 CB8	TM12 CB7	TM12 CB6	TM12 CB5	TM12 CB4	TM12 CB3	TM12 CB2	TM12 CB1	TM12 CB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM12CB : x'00FEC8'

Timer 12 Compare/ Capture Register B
16-bit access register

TM12CB sets the timer 12 PWM duty, changes PWM and generates a timer 12 capture B interrupt.

When capture is selected, this register is read only. A timer 12 capture B interrupt is generated when capture occurs. When compare is selected, set the PWM duty. When this register matches the timer 12 binary counter, a timer 12 capture B interrupt occurs.

This register write only 16-bit data. Use the MOV instruction to set the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
T12C BX15	T12C BX14	T12C BX13	T12C BX12	T12C BX11	T12C BX10	T12C BX9	T12C BX8	T12C BX7	T12C BX6	T12C BX5	T12C BX4	T12C BX3	T12C BX2	T12C BX1	T12C BX0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-

TM12CBX : x'00FECA'

Timer 12 Compare/ Capture Register Set B
16-bit access register

This register is valid only when the associated compare register is set to the double-buffer mode. TM12CBX cannot read or write. The contents of TM12CB are loaded to TM12CBX by write signal.

TM12CBX sets the PWM cycle. When TM12BC=TM12CBX, a timer 12 capture B interrupt occurs. The contents of TM12CB are loaded to TM12CBX by a timer 12 capture B interrupt and TM12CBX prevents the PWM loss.

This register writes only 16-bit data. Use the MOV instruction to set the data.

7	6	5	4	3	2	1	0
TM13 CA7	TM13 CA6	TM13 CA5	TM13 CA4	TM13 CA3	TM13 CA2	TM13 CA1	TM13 CA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM13CA : x'00FE0A'

Timer 13 Capture A Register

8-bit access register

TM13CA sets the timing of changing the PWM waveform output from TM13OA pin from low level to high level.

The valid range for TM13CA is 1 to TM13BR.

7	6	5	4	3	2	1	0
TM13 CB7	TM13 CB6	TM13 CB5	TM13 CB4	TM13 CB3	TM13 CB2	TM13 CB1	TM13 CB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM13CB : x'00FE1A'

Timer 13 Capture B Register

8-bit access register

TM13CB sets the timing of changing the PWM waveform output from TM13OB pin from low level to high level.

The valid range for TM13CB is 1 to TM13BR.

7	6	5	4	3	2	1	0
TM14 CA7	TM14 CA6	TM14 CA5	TM14 CA4	TM14 CA3	TM14 CA2	TM14 CA1	TM14 CA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM14CA : x'00FE0B'

Timer 14 Capture A Register

8-bit access register

TM14CA sets the timing of changing the PWM waveform output from TM14OA pin from low level to high level.

The valid range for TM14CA is 1 to TM14BR.

7	6	5	4	3	2	1	0
TM14 CB7	TM14 CB6	TM14 CB5	TM14 CB4	TM14 CB3	TM14 CB2	TM14 CB1	TM14 CB0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM14CB : x'00FE1B'

Timer 14 Capture B Register

8-bit access register

TM14CB sets the timing of changing the PWM waveform output from TM14OB pin from low level to high level.

The valid range for TM14CB is 1 to TM14BR.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
TM15 CA15	TM15 CA14	TM15 CA13	TM15 CA12	TM15 CA11	TM15 CA10	TM15 CA9	TM15 CA8	TM15 CA7	TM15 CA6	TM15 CA5	TM15 CA4	TM15 CA3	TM15 CA2	TM15 CA1	TM15 CA0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

TM15CA : x'00FED4'

Timer 15 Capture Register A

16-bit access register

TM15CA captures the contents of TM15BC on the rising of TM15IA pin.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	AN NCH2	AN NCH1	AN NCH0	-	AN 1CH2	AN 1CH1	AN 1CH0	AN EN	AN TC	AN DEC	-	AN CK1	AN CK0	AN MD1	AN MD0
R	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0/1	0/1	0/1	0	0/1	0/1	0/1	0/1	0/1	0/1	0	0/1	0/1	0/1	0/1

ANCTR : x'00FF00'

A/D Converter Control Register

16-bit access register

ANCTR sets the A/D converter operating conditions.

14:12 Channel Selection for Multiple Channel Conversion

000: Convert AN0
001: Convert from AN0 to AN1
010: Convert from AN0 to AN2
011: Convert from AN0 to AN3
100: Convert from AN0 to AN4
101: Convert from AN0 to AN5
110: Convert from AN0 to AN6
111: Convert from AN0 to AN7

10:8 Channel Selection for Single Channel Conversion

000: Convert AN0
001: Convert AN1
010: Convert AN2
011: Convert AN3
100: Convert AN4
101: Convert AN5
110: Convert AN6
111: Convert AN7

7 Conversion Start/Execution Flag

0: No conversion
1: Conversion in progress

6 Conversion Start at Timer 3 underflow

0: Disable
1: Enable

5 AD Converter Resolution

0: 8-bit
1: 10-bit

3:2 Clock Source Selection

00: BOSC/2
01: BOSC/4
10: BOSC/8
11: BOSC/16

1:0 Operating Mode Selection

00: Single channel, single conversion
01: Multiple channels, single conversion
10: Single channel, continuous conversion
11: Multiple channels, continuous conversion

AN0BUF : x'00FF08'

AN0 Conversion Data Buffer

16-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	AN0 BUF9	AN0 BUF8	AN0 BUF7	AN0 BUF6	AN0 BUF5	AN0 BUF4	AN0 BUF3	AN0 BUF2	AN0 BUF1	AN0 BUF0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
0	0	0	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AN0 conversion data

When 8-bit is selected as A/D converter resolution, the AN0BUF[7:0] bits hold the data and the AN0BUF[9:8] bits become 0. When 10- bit is selected as A/D converter resolution, the AN0BUF[9:0] bits hold the data.

AN1BUF : x'00FF0A'

AN1 Conversion Data Buffer

16-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	AN1 BUF9	AN1 BUF8	AN1 BUF7	AN1 BUF6	AN1 BUF5	AN1 BUF4	AN1 BUF3	AN1 BUF2	AN1 BUF1	AN1 BUF0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
0	0	0	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AN1 conversion data

When 8-bit is selected as A/D converter resolution, the AN1BUF[7:0] bits hold the data and the AN1BUF[9:8] bits become 0. When 10- bit is selected as A/D converter resolution, the AN1BUF[9:0] bits hold the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	AN2 BUF9	AN2 BUF8	AN2 BUF7	AN2 BUF6	AN2 BUF5	AN2 BUF4	AN2 BUF3	AN2 BUF2	AN2 BUF1	AN2 BUF0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
0	0	0	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AN2BUF : x'00FF0C'

AN2 Conversion Data Buffer

16-bit access register

AN2 conversion data

When 8-bit is selected as A/D converter resolution, the AN2BUF[7:0] bits hold the data and the AN2BUF[9:8] bits become 0. When 10- bit is selected as A/D converter resolution, the AN2BUF[9:0] bits hold the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	AN3 BUF9	AN3 BUF8	AN3 BUF7	AN3 BUF6	AN3 BUF5	AN3 BUF4	AN3 BUF3	AN3 BUF2	AN3 BUF1	AN3 BUF0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
0	0	0	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AN3BUF : x'00FF0E'

AN3 Conversion Data Buffer

16-bit access register

AN3 conversion data

When 8-bit is selected as A/D converter resolution, the AN3BUF[7:0] bits hold the data and the AN3BUF[9:8] bits become 0. When 10- bit is selected as A/D converter resolution, the AN3BUF[9:0] bits hold the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	AN4 BUF9	AN4 BUF8	AN4 BUF7	AN4 BUF6	AN4 BUF5	AN4 BUF4	AN4 BUF3	AN4 BUF2	AN4 BUF1	AN4 BUF0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
0	0	0	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AN4BUF : x'00FF10'

AN4 Conversion Data Buffer

16-bit access register

AN4 conversion data

When 8-bit is selected as A/D converter resolution, the AN4BUF[7:0] bits hold the data and the AN4BUF[9:8] bits become 0. When 10- bit is selected as A/D converter resolution, the AN4BUF[9:0] bits hold the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	AN4 BUF9	AN4 BUF8	AN4 BUF7	AN4 BUF6	AN4 BUF5	AN4 BUF4	AN4 BUF3	AN4 BUF2	AN4 BUF1	AN4 BUF0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
0	0	0	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AN5BUF : x'00FF12'

AN5 Conversion Data Buffer

16-bit access register

AN5 conversion data

When 8-bit is selected as A/D converter resolution, the AN5BUF[7:0] bits hold the data and the AN5BUF[9:8] bits become 0. When 10- bit is selected as A/D converter resolution, the AN5BUF[9:0] bits hold the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	AN6 BUF9	AN6 BUF8	AN6 BUF7	AN6 BUF6	AN6 BUF5	AN6 BUF4	AN6 BUF3	AN6 BUF2	AN6 BUF1	AN6 BUF0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
0	0	0	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AN6BUF : x'00FF14'

AN6 Conversion Data Buffer

16-bit access register

AN6 conversion data

When 8-bit is selected as A/D converter resolution, the AN6BUF[7:0] bits hold the data and the AN6BUF[9:8] bits become 0. When 10- bit is selected as A/D converter resolution, the AN6BUF[9:0] bits hold the data.

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	AN7 BUF9	AN7 BUF8	AN7 BUF7	AN7 BUF6	AN7 BUF5	AN7 BUF4	AN7 BUF3	AN7 BUF2	AN7 BUF1	AN7 BUF0
R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R
0	0	0	0	0	0	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

AN7BUF : x'00FF16'

AN7 Conversion Data Buffer

16-bit access register

AN7 conversion data

When 8-bit is selected as A/D converter resolution, the AN7BUF[7:0] bits hold the data and the AN7BUF[9:8] bits become 0. When 10- bit is selected as A/D converter resolution, the AN7BUF[9:0] bits hold the data.

DA0CTR : x'00FF40'

DA0 Converter Control Register

8-bit access register

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	DA0 ON
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

DA0CTR controls DA0 conversion.

0	**DA0 Conversion Start**	0: Disable 1: Enable

DA0BUF : x'00FF41'

DA0 Conversion Data Buffer

8-bit access register (16-bit access is possible from even address)

7	6	5	4	3	2	1	0
DA0 BUF7	DA0 BUF6	DA0 BUF5	DA0 BUF4	DA0 BUF3	DA0 BUF2	DA0 BUF1	DA0 BUF0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

DA0 conversion data

DA1CTR : x'00FF42'

DA1 Converter Control Register

8-bit access register

7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	DA1 ON
R	R	R	R	R	R	R	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0/1

DA1CTR controls DA1 conversion.

0	**DA1 Conversion Start**	0: Disable 1: Enable

DA1BUF : x'00FF43'

DA1 Conversion Data Buffer

8-bit access register (16-bit access is possible from even address)

7	6	5	4	3	2	1	0
DA1 BUF7	DA1 BUF6	DA1 BUF5	DA1 BUF4	DA1 BUF3	DA1 BUF2	DA1 BUF1	DA1 BUF0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

DA1 conversion data

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EW 33	EW 32	EW 31	EW 30	EW 23	EW 22	EW 21	EW 20	EW 13	EW 12	EW 11	EW 10	EW 03	EW 02	EW 01	EW 00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

EXWMD : x'00FF80'

External Memory Wait Register

16-bit access register

EXWMD sets the external memory wait cycles.

* Please refer to Figure 2-1-1 Address Space on page 52 for address allocation of external memory spaces.

15:12 Wait Setting for External Memory Space 3 *

11:8 Wait Setting for External Memory Space 2 *

7:4 Wait Setting for External Memory Space 1 *

3:0 Wait Setting for External Memory Space 0 *

Setting		Waits	Cycles
0	0000	0.0	1.0
1	0001	0.5	1.5
2	0010	1.0	2.0
3	0011	1.5	2.5
4	0100	2.0	3.0
5	0101	2.5	3.5
6	0110	3.0	4.0
7	0111	3.5	4.5
8	1000	4.0	5.0
9	1001	4.5	5.5
10	1010	5.0	6.0
11	1011	5.5	6.5
12	1100	6.0	7.0
13	1101	6.5	7.5
14	1110	7.0	8.0
15	1111	perform handshake mode by WAIT pin	

0.5 wait cycle corresponds to BOSC 1 cycle. 1 wait cycle corresponds to 1 cycle of instruction. With a 34-MHz oscillator,

0.5 wait cycle = 29.4 ns

1 wait cycle = 58.8 ns

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EB 31	EB 30	EB 21	EB 20	EB 11	EB 10	EB 01	EB 00	BRS 1	BRS 0	BRC 3	BRC 2	BRC 1	BRC 0	IOW 1	IOW 0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Single-chip Mode	0	0	0	0	0	0	0	undefined	0	0	0	0	0	0	1	1
Memory Expansion Mode (16-bit bus width)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
Memory Expansion Mode (8-bit bus width)	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1
Processor Mode (16-bit bus width)	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
Processor Mode (8-bit bus width)	0	0	0	0	0	0	0	1	0	0	0	0	0	0	1	1
	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

MEMMD1 : x'00FF82'

Memory Mode Setup 1 Register

16-bit access register

MEMMD1 sets the bus width for external memory and the wait cycles for internal I/O registers.

Do not access the burst ROM area and other areas consecutively.

15:14 Bus Width Setting for External Memory Space 3 *

13:12 Bus Width Setting for External Memory Space 2 *

11:10 Bus Width Setting for External Memory Space 1 *

9:8 Bus Width Setting for External Memory Space 0 *

00: 16-bit bus width
01: 8-bit bus width
10: Reserved
11: 8-bit bus width when A8 is high,
16-bit bus width when A8 is low

* Please refer to Figure 2-1-1 Address Space on page 52 for address allocation of external memory spaces.

EB[01:00] bits at reset can be changed depending on $\overline{\text{WORD}}$ pin input.

7:6 Address Setting for Burst Operation

	16-bit Bus Width	8-bit Bus Width
00: A0, A1	2 words	4 bytes
01: A0, A1, A2	4 words	8 bytes
10: A0, A1, A2, A3	8 words	16 bytes
11: A0, A1, A2, A3, A4	16 words	32 bytes

5 Burst ROM setting for External Memory Space 3

0: Disable
1: Enable

4 Burst ROM setting for External Memory Space 2

0: Disable
1: Enable

3 Burst ROM setting for External Memory Space 1

0: Disable
1: Enable

2 Burst ROM setting for External Memory Space 0

0: Disable
1: Enable

1:0 Wait Setting for Internal I/O Space

00: 1.0 wait cycle
01: 1.5 wait cycles
10: 2.0 wait cycles
11: 3.0 wait cycles

0.5 wait cycle corresponds to BOSC 1 cycle. 1 wait cycle corresponds to 1 cycle of instruction. With a 34-MHz oscillator,
0.5 wait cycle = 29.4 ns
1 wait cycle = 58.8 ns

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	HS2	HS1	HS0	-	-	reserved	reserved	-	BST2	BST1	BST0
R	R	R	R	R	R/W	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0/1	0/1	0/1	0	0	0/1	0/1	0	0/1	0/1	0/1

MEMMD2 : x'00FF84'

Memory Mode Setup 2 Register

16-bit access register

MEMMD2 sets the burst ROM cycles and changes the pulse timing of $\overline{\text{WEH}}$, $\overline{\text{WEL}}$ and $\overline{\text{RE}}$.

10:8 Fixed Wait Setting

000: No wait
001: 0.5 wait cycle
010: 1 wait cycle
011: 1.5 wait cycles
100: 2 wait cycles
101: 2.5 wait cycles
110: 3 wait cycles
111: 3.5 wait cycles

5 Reserved

Set to 0

4 Reserved

Set to 0

2:0 Cycle Setting for Burst ROM Shortening (First Cycle at Burst Access)

000: 0.5 cycle
001: 1 cycle
010: 1.5 cycles
011: 2 cycles
100: 2.5 cycles
101: 3 cycles
110: 3.5 cycles
111: 4 cycles

M

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ARE 3	ARE 2	ARE 1	ARE 0	MMD 1	MMD 0	ASEN	SEL 2	SEL 1	SEL 0	CAS 2	CAS 1	CAS 0	RAS 2	RAS 1	RAS 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

DRAMMD1 : x'00FF90'

DRAM Control 1 Register

16-bit access register

DRAMMD1 sets the DRAM modes.

* Please refer to Figure 2-1-1 Address Space on page 52 for address allocation of external memory spaces.

15 **DRAM Operation for External Memory Space 3 ***
0: Disable
1: Enable

14 **DRAM Operation for External Memory Space 2 ***
0: Disable
1: Enable

13 **DRAM Operation for External Memory Space 1 ***
0: Disable
1: Enable

12 **DRAM Operation for External Memory Space 0 ***
0: Disable
1: Enable

11:10 **Shift Size of DRAM Address**
00: 8-bit
01: 9-bit
10: 10-bit
11: Reserved

MMD(1:0) Setting		00 Shift 8	01 Shift 9	10 Shift 10
Pin Name	ROW Address Output	COL Address Output		
P46	A22	-	(A11)	(A11)
P45	A21	-	(A10)	(A10)
P44	A20	-	(Lo)	A10
P43	A19	-	(Lo)	A9
P42	A18	-	A9	A8
P41	A17	-	A8	A7
P40	A16	A8	A7	A6
P37	A15	A7	A6	A5
P36	A14	A6	A5	A4
P35	A13	A5	A4	A3
P34	A12	A4	A3	A2
P33	A11	A3	A2	A1
P32	A10	A2	A1	A0
P31	A9	A1	A0	(A0)
P30	A8	A0	-	-

9 **Shift Setting from Row addresses of AD15-AD0 pins to Column addresses**
0: Don't shift
1: Shift

8:6 **Shift Timing Setting from Row Address to Column Address**

5:3 **Timing Setting of $\overline{\text{CAS}}$ Falling Edge**

2:0 **Timing Setting of $\overline{\text{RAS}}$ Falling Edge**

000: At the beginning of 0.5 cycle
001: At the beginning of 1.0 cycle
010: At the beginning of 1.5 cycles
011: At the beginning of 2.0 cycles
100: At the beginning of 2.5 cycles
101: At the beginning of 3.0 cycles
110: At the beginning of 3.5 cycles
111: At the beginning of 4.0 cycles

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
DRAM ACC	DRAM TM	reserv ed	reserv ed	reserv ed	RON	RCY 3	RCY 2	RCY 1	RCY 0	RCS 2	RCS 1	RCS 0	RRS 2	RRS 1	RRS 0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

DRAMMD2 : x'00FF92'

DRAM Control 2 Register

16-bit access register

DRAMMD2 sets the DRAM modes.

Use only 2WE method in DRAM mode. Do not use 2CAS method.

* DRAM is refreshed once when a timer 10 or 12 underflow interrupt occurs. At 256 times/8 ms, the refresh interval is 31.25 µs or less.

15 DRAM Access Method Selection
0: 2WE method
1: Reserved

14 Clock Source Selection for DRAM Refresh
0: Timer 12 underflow
1: Timer 10 underflow

13:11 Reserved
Set to 0

10 DRAM Refresh Enable
0: Disable
1: Enable *

9:6 Cycle Setting at Refresh
0000: 2.0 cycles
0001: 2.5 cycles
0010: 3.0 cycles
0011: 3.5 cycles
0100: 4.0 cycles
0101: 4.5 cycles
0110: 5.0 cycles
0111: 5.5 cycles
1000: 6.0 cycles
1001: 6.5 cycles
1010: 7.0 cycles
Other: 7.0 cycles

5:3 Timing Setting of $\overline{\text{CAS}}$ Falling Edge

2:0 Timing Setting of $\overline{\text{RAS}}$ Falling Edge

000: At the beginning of 0.5 cycle
001: At the beginning of 1.0 cycle
010: At the beginning of 1.5 cycles
011: At the beginning of 2.0 cycles
100: At the beginning of 2.5 cycles
101: At the beginning of 3.0 cycles
110: At the beginning of 3.5 cycles
111: At the beginning of 4.0 cycles

REEDGE : x'00FF86'

$\overline{\text{RE}}$ Waveform Control Register

16-bit access register

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	REES 31	REES 30	REEL 31	REEL 30	REES 21	REES 20	REEL 21	REEL 20	REES 11	REES 10	REEL 11	REEL 10	REES 01	REES 00	REEL 01	REEL 00
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Processor Mode	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
Other Modes	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

REEDGE sets the $\overline{\text{RE}}$ waveform control modes.

The $\overline{\text{RE}}$ short mode and the $\overline{\text{RE}}$ late mode do not affect the $\overline{\text{BSTRE}}$ pin connecting burst ROM.

15:14 $\overline{\text{RE}}$ Short Mode in $\overline{\text{CS3}}$ Space

00: $\overline{\text{RE}}$ Short 0 Mode (Reset)
01: $\overline{\text{RE}}$ Short 0.5 Mode
10: $\overline{\text{RE}}$ Short 1 Mode
11: $\overline{\text{RE}}$ Short 1.5 Mode

13:12 $\overline{\text{RE}}$ Late Mode in $\overline{\text{CS3}}$ Space

00: $\overline{\text{RE}}$ Late 0.5 Mode (Reset)
01: $\overline{\text{RE}}$ Late 1 Mode
10: $\overline{\text{RE}}$ Late 2 Mode
11: $\overline{\text{RE}}$ Late 3 Mode

11:10 $\overline{\text{RE}}$ Short Mode in $\overline{\text{CS2}}$ Space

00: $\overline{\text{RE}}$ Short 0 Mode (Reset)
01: $\overline{\text{RE}}$ Short 0.5 Mode
10: $\overline{\text{RE}}$ Short 1 Mode
11: $\overline{\text{RE}}$ Short 1.5 Mode

9:8 $\overline{\text{RE}}$ Late Mode in $\overline{\text{CS2}}$ Space

00: $\overline{\text{RE}}$ Late 0.5 Mode (Reset)
01: $\overline{\text{RE}}$ Late 1 Mode
10: $\overline{\text{RE}}$ Late 2 Mode
11: $\overline{\text{RE}}$ Late 3 Mode

7:6 $\overline{\text{RE}}$ Short Mode in $\overline{\text{CS1}}$ Space

00: $\overline{\text{RE}}$ Short 0 Mode (Reset)
01: $\overline{\text{RE}}$ Short 0.5 Mode
10: $\overline{\text{RE}}$ Short 1 Mode
11: $\overline{\text{RE}}$ Short 1.5 Mode

5:4 $\overline{\text{RE}}$ Late Mode in $\overline{\text{CS1}}$ Space

00: $\overline{\text{RE}}$ Late 0.5 Mode (Reset)
01: $\overline{\text{RE}}$ Late 1 Mode
10: $\overline{\text{RE}}$ Late 2 Mode
11: $\overline{\text{RE}}$ Late 3 Mode

3:2 $\overline{\text{RE}}$ Short Mode in $\overline{\text{CS0}}$ Space

00: $\overline{\text{RE}}$ Short 0 Mode
01: $\overline{\text{RE}}$ Short 0.5 Mode (Reset)
10: $\overline{\text{RE}}$ Short 1 Mode
11: $\overline{\text{RE}}$ Short 1.5 Mode

1:0 $\overline{\text{RE}}$ Late Mode in CS0 Space

00: $\overline{\text{RE}}$ Late 0.5 Mode (Reset)
01: $\overline{\text{RE}}$ Late 1 Mode
10: $\overline{\text{RE}}$ Late 2 Mode
11: $\overline{\text{RE}}$ Late 3 Mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
WEES 31	WEES 30	WEEL 31	WEEL 30	WEES 21	WEES 20	WEEL 21	WEEL 20	WEES 11	WEES 10	WEEL 11	WEEL 10	WEES 01	WEES 00	WEEL 01	WEEL 00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

WEEDGE : x'00FF88'

$\overline{WE}$ Waveform Control Register

16-bit access register

WEEDGE sets the $\overline{WE}$ waveform control modes.

15:14 $\overline{WE}$ Short Mode in $\overline{CS3}$ Space

00: $\overline{WE}$ Short 0 Mode (Reset)
01: $\overline{WE}$ Short 0.5 Mode
10: $\overline{WE}$ Short 1 Mode
11: $\overline{WE}$ Short 1.5 Mode

13:12 $\overline{WE}$ Late Mode in $\overline{CS3}$ Space

00: $\overline{WE}$ Late 1 Mode (Reset)
01: $\overline{WE}$ Late 2 Mode
10: $\overline{WE}$ Late 3 Mode
11: Reserved

11:10 $\overline{WE}$ Short Mode in $\overline{CS2}$ Space

00: $\overline{WE}$ Short 0 Mode (Reset)
01: $\overline{WE}$ Short 0.5 Mode
10: $\overline{WE}$ Short 1 Mode
11: $\overline{WE}$ Short 1.5 Mode

9:8 $\overline{WE}$ Late Mode in $\overline{CS2}$ Space

00: $\overline{WE}$ Late 1 Mode (Reset)
01: $\overline{WE}$ Late 2 Mode
10: $\overline{WE}$ Late 3 Mode
11: Reserved

7:6 $\overline{WE}$ Short Mode in $\overline{CS1}$ Space

00: $\overline{WE}$ Short 0 Mode (Reset)
01: $\overline{WE}$ Short 0.5 Mode
10: $\overline{WE}$ Short 1 Mode
11: $\overline{WE}$ Short 1.5 Mode

5:4 $\overline{WE}$ Late Mode in $\overline{CS1}$ Space

00: $\overline{WE}$ Late 1 Mode (Reset)
01: $\overline{WE}$ Late 2 Mode
10: $\overline{WE}$ Late 3 Mode
11: Reserved

3:2 $\overline{WE}$ Short Mode in $\overline{CS0}$ Space

00: $\overline{WE}$ Short 0 Mode (Reset)
01: $\overline{WE}$ Short 0.5 Mode
10: $\overline{WE}$ Short 1 Mode
11: $\overline{WE}$ Short 1.5 Mode

1:0 $\overline{WE}$ Late Mode in $\overline{CS0}$ Space

00: $\overline{WE}$ Late 1 Mode (Reset)
01: $\overline{WE}$ Late 2 Mode
10: $\overline{WE}$ Late 3 Mode
11: Reserved

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
ALEG 31	ALEG 30	ALEL 31	ALEL 30	ALEG 21	ALEG 20	ALEL 21	ALEL 20	ALEG 11	ALEG 10	ALEL 11	ALEL 10	ALEG 01	ALEG 00	ALEL 01	ALEL 00
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

ALEEDGE : x'00FF8A'

ALE Waveform Control Register

16-bit access register

ALEEDGE sets the ALE waveform control modes.

15:14 ALE Long Mode in $\overline{CS3}$ Space
00: ALE Long 0 Mode (Reset)
01: ALE Long 0.5 Mode
10: ALE Long 1 Mode
11: ALE Long 1.5 Mode

13:12 ALE Late Mode in $\overline{CS3}$ Space
00: ALE Late 0 Mode (Reset)
01: ALE Late 0.5 Mode
10: ALE Late 1 Mode
11: ALE Late 1.5 Mode

11:10 ALE Long Mode in $\overline{CS2}$ Space
00: ALE Long 0 Mode (Reset)
01: ALE Long 0.5 Mode
10: ALE Long 1 Mode
11: ALE Long 1.5 Mode

9:8 ALE Late Mode in $\overline{CS2}$ Space
00: ALE Late 0 Mode (Reset)
01: ALE Late 0.5 Mode
10: ALE Late 1 Mode
11: ALE Late 1.5 Mode

7:6 ALE Long Mode in $\overline{CS1}$ Space
00: ALE Long 0 Mode (Reset)
01: ALE Long 0.5 Mode
10: ALE Long 1 Mode
11: ALE Long 1.5 Mode

5:4 ALE Late Mode in $\overline{CS1}$ Space
00: ALE Late 0 Mode (Reset)
01: ALE Late 0.5 Mode
10: ALE Late 1 Mode
11: ALE Late 1.5 Mode

3:2 ALE Long Mode in $\overline{CS0}$ Space
00: ALE Long 0 Mode
01: ALE Long 0.5 Mode
10: ALE Long 1 Mode (Reset)
11: ALE Long 1.5 Mode

1:0 ALE Late Mode in $\overline{CS0}$ Space
00: ALE Late 0 Mode
01: ALE Late 0.5 Mode
10: ALE Late 1 Mode (Reset)
11: ALE Late 1.5 Mode

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	ADL 31	ADL 30	ADL 21	ADL 20	ADL 11	ADL 10	ADL 01	ADL 00
R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

MPXADR : x'00FF8C'

Address Output Time Control Register

16-bit access register

MPXADR sets the address output time control modes during address/data shared mode.

7:6 Address Long Mode in $\overline{\text{CS3}}$ Space

00: AD Long 1 Mode (Reset)
01: AD Long 1.5 Mode
10: AD Long 2 Mode
11: AD Long 3 Mode

5:4 Address Long Mode in $\overline{\text{CS2}}$ Space

00: AD Long 1 Mode (Reset)
01: AD Long 1.5 Mode
10: AD Long 2 Mode
11: AD Long 3 Mode

3:2 Address Long Mode in $\overline{\text{CS1}}$ Space

00: AD Long 1 Mode (Reset)
01: AD Long 1.5 Mode
10: AD Long 2 Mode
11: AD Long 3 Mode

1:0 Address Long Mode in $\overline{\text{CS0}}$ Space

00: AD Long 1 Mode
01: AD Long 1.5 Mode
10: AD Long 2 Mode
11: AD Long 3 Mode (Reset)

PBSW : x'00FFBC'

Pointer Byte Swap Register

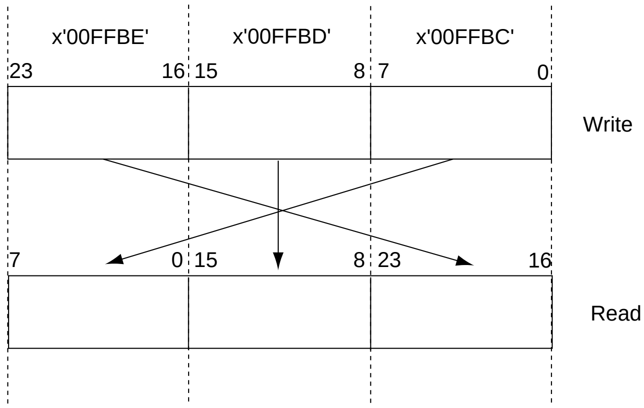
16/24-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
PBS W15	PBS W14	PBS W13	PBS W12	PBS W11	PBS W10	PBS W9	PBS W8	PBS W7	PBS W6	PBS W5	PBS W4	PBS W3	PBS W2	PBS W1	PBS W0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
-	-	-	-	-	-	-	-	PBS W23	PBS W22	PBS W21	PBS W20	PBS W19	PBS W18	PBS W17	PBS W16
R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

PBSW writes 24-bit pointer data.

During read operations, the upper 8-bit data and the lower 8-bit data are inverted. The middle 8-bit remains.



LBSWL : x'00FFCC'

Long Word Byte Swap Register L

8/16-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LBS WL15	LBS WL14	LBS WL13	LBS WL12	LBS WL11	LBS WL10	LBS WL9	LBS WL8	LBS WL7	LBS WL6	LBS WL5	LBS WL4	LBS WL3	LBS WL2	LBS WL1	LBS WL0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

LBSWL writes 16-bit data.

During read operations, bits [7:0] read bits[15:8] of the LBSWH register, and bits 15-8 read bits[7:0] of the LBSWH register. Combining with the LBSWH register, 24-bit upper and lower data can swapped in 8-bit unit. In addition, 16-bit upper and lower data can be swapped by writing the 16-bit data to the LBSWL register and reading the data from the LBSWH register.

LBSWH : x'00FFCE'

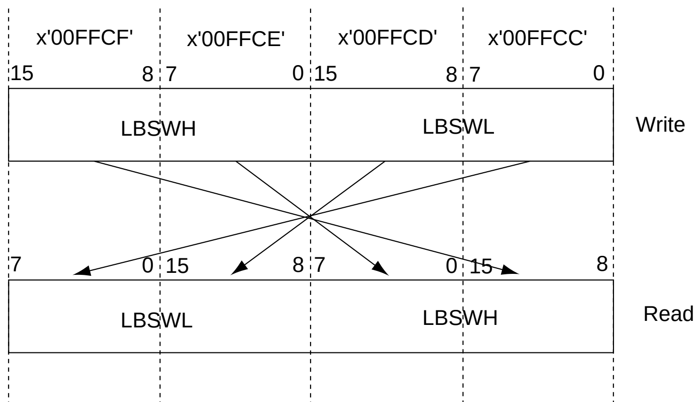
Long Word Byte Swap Register H

8/16-bit access register

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
LBS WH15	LBS WH14	LBS WH13	LBS WH12	LBS WH11	LBS WH10	LBS WH9	LBS WH8	LBS WH7	LBS WH6	LBS WH5	LBS WH4	LBS WH3	LBS WH2	LBS WH1	LBS WH0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

LBSWH writes 16-bit data.

During read operations, bits[7:0] read bits[15:8] of the LBSWL register, and bits[15:8] read bits [7:0] of the LBSWL register.



P0PLU : x'00FFB0'

Port 0 Pullup Control Register

8-bit access register

P0PLU controls the port 0 pullup resistor.

7	6	5	4	3	2	1	0
P0 PLU7	P0 PLU6	P0 PLU5	P0 PLU4	P0 PLU3	P0 PLU2	P0 PLU1	P0 PLU0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 Port 0 Pullup Resistor 0: Off
1: On

P1PLU : x'00FFB1'

Port 1 Pullup Control Register

8-bit access register

P1PLU controls the port 1 pullup resistor.

7	6	5	4	3	2	1	0
P1 PLU7	P1 PLU6	P1 PLU5	P1 PLU4	P1 PLU3	P1 PLU2	P1 PLU1	P1 PLU0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 Port 1 Pullup Resistor 0: Off
1: On

P2PLU : x'00FFB2'

Port 2 Pullup Control Register

8-bit access register

P2PLU controls the port 2 pullup resistor.

7	6	5	4	3	2	1	0
P2 PLU7	P2 PLU6	P2 PLU5	P2 PLU4	P2 PLU3	P2 PLU2	P2 PLU1	P2 PLU0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 Port 2 Pullup Resistor 0: Off
1: On

P3PLU : x'00FFB3'

Port 3 Pullup Control Register

8-bit access register

P3PLU controls the port 3 pullup resistor.

7	6	5	4	3	2	1	0
P3 PLU7	P3 PLU6	P3 PLU5	P3 PLU4	P3 PLU3	P3 PLU2	P3 PLU1	P3 PLU0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 Port 3 Pullup Resistor 0: Off
1: On

P4PLU : x'00FFB4'
Port 4 Pullup Control Register

8-bit access register

7	6	5	4	3	2	1	0
P4 PLU7	P4 PLU6	P4 PLU5	P4 PLU4	P4 PLU3	P4 PLU2	P4 PLU1	P4 PLU0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

P4PLU controls the port 4 pullup resistor.

7:0 **Port 4 Pullup Resistor** 0: Off
1: On

P5PLU : x'00FFB5'
Port 5 Pullup Control Register

8-bit access register

7	6	5	4	3	2	1	0
P5 PLU7	P5 PLU6	P5 PLU5	P5 PLU4	P5 PLU3	P5 PLU2	P5 PLU1	P5 PLU0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

P5PLU controls the port 5 pullup resistor.

7:0 **Port 5 Pullup Resistor** 0: Off
1: On

P6PLU : x'00FFB6'
Port 6 Pullup Control Register

8-bit access register

7	6	5	4	3	2	1	0
-	-	-	-	P6 PLU3	P6 PLU2	P6 PLU1	P6 PLU0
R	R	R	R	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0/1	0/1	0/1	0/1

P6PLU controls the port 6 pullup resistor.

3:0 **Port 6 Pullup Resistor** 0: Off
1: On

P

P7PLU : x'00FFB7'
Port 7 Pullup Control Register

8-bit access register

7	6	5	4	3	2	1	0
-	-	P7 PLU5	P7 PLU4	P7 PLU3	P7 PLU2	P7 PLU1	P7 PLU0
R	R	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0/1	0/1	0/1	0/1	0/1	0/1

P7PLU controls the port 7 pullup resistor.

5:0 **Port 7 Pullup Resistor** 0: Off
1: On

P8PLU : x'00FFB8'

Port 8 Pullup Control Register

8-bit access register

P8PLU controls the port 8 pullup resistor.

7	6	5	4	3	2	1	0
P8 PLU7	P8 PLU6	P8 PLU5	P8 PLU4	P8 PLU3	P8 PLU2	P8 PLU1	P8 PLU0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 **Port 8 Pullup Resistor** 0: Off
1: On

P9PLU : x'00FFB9'

Port 9 Pullup Control Register

8-bit access register

P9PLU controls the port 9 pullup resistor.

7	6	5	4	3	2	1	0
P9 PLU7	P9 PLU6	P9 PLU5	P9 PLU4	P9 PLU3	P9 PLU2	P9 PLU1	P9 PLU0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 **Port 9 Pullup Resistor** 0: Off
1: On

PAPLU : x'00FFBA'

Port A Pullup Control Register

8-bit access register

PAPLU controls the port A pullup resistor.

7	6	5	4	3	2	1	0
-	-	PA PLU5	PA PLU4	PA PLU3	PA PLU2	PA PLU1	PA PLU0
R	R	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0/1	0/1	0/1	0/1	0/1	0/1

5:0 **Port A Pullup Resistor** 0: Off
1: On

PBPLU : x'00FFBB'

Port B Pullup Control Register

8-bit access register

PBPLU controls the port B pullup resistor.

7	6	5	4	3	2	1	0
-	-	-	reserved	reserved	reserved	PB PLU1	PB PLU0
R	R	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0/1	0/1	0/1	0/1	0/1	0/1

4:2 **Reserved** Set to 0

1:0 **Port B Pullup Resistor** 0: Off
1: On

P0OUT : x'00FFC0'

Port 0 Output Register

8-bit access register

7	6	5	4	3	2	1	0
P0 OUT7	P0 OUT6	P0 OUT5	P0 OUT4	P0 OUT3	P0 OUT2	P0 OUT1	P0 OUT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 **Port 0 Output** 0: Output low
1: Output high

P0OUT sets the data output to the port 0.

P1OUT : x'00FFC1'

Port 1 Output Register

8-bit access register

7	6	5	4	3	2	1	0
P1 OUT7	P1 OUT6	P1 OUT5	P1 OUT4	P1 OUT3	P1 OUT2	P1 OUT1	P1 OUT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 **Port 1 Output** 0: Output low
1: Output high

P1OUT sets the data output to the port 1.

P2OUT : x'00FFC2'

Port 2 Output Register

8-bit access register

7	6	5	4	3	2	1	0
P2 OUT7	P2 OUT6	P2 OUT5	P2 OUT4	P2 OUT3	P2 OUT2	P2 OUT1	P2 OUT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 **Port 2 Output** 0: Output low
1: Output high

P2OUT sets the data output to the port 2.

P3OUT : x'00FFC3'

Port 3 Output Register

8-bit access register

7	6	5	4	3	2	1	0
P3 OUT7	P3 OUT6	P3 OUT5	P3 OUT4	P3 OUT3	P3 OUT2	P3 OUT1	P3 OUT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 **Port 3 Output** 0: Output low
1: Output high

P3OUT sets the data output to the port 3.

P4OUT : x'00FFC4'

Port 4 Output Register

8-bit access register

7	6	5	4	3	2	1	0
P4 OUT7	P4 OUT6	P4 OUT5	P4 OUT4	P4 OUT3	P4 OUT2	P4 OUT1	P4 OUT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

P4OUT sets the data output to the port 4.

7:0 **Port 4 Output** 0: Output low
1: Output high

P5OUT : x'00FFC5'

Port 5 Output Register

8-bit access register

7	6	5	4	3	2	1	0
P5 OUT7	P5 OUT6	P5 OUT5	P5 OUT4	P5 OUT3	P5 OUT2	P5 OUT1	P5 OUT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

P5OUT sets the data output to the port 5.

7:0 **Port 5 Output** 0: Output low
1: Output high

P6OUT : x'00FFC6'

Port 6 Output Register

8-bit access register

7	6	5	4	3	2	1	0
-	-	-	-	P6 OUT3	P6 OUT2	P6 OUT1	P6 OUT0
R	R	R	R	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0/1	0/1	0/1	0/1

P6OUT sets the data output to the port 6.

3:0 **Port 6 Output** 0: Output low
1: Output high

P7OUT : x'00FFC7'

Port 7 Output Register

8-bit access register

7	6	5	4	3	2	1	0
-	-	P7 OUT5	P7 OUT4	P7 OUT3	P7 OUT2	P7 OUT1	P7 OUT0
R	R	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0/1	0/1	0/1	0/1	0/1	0/1

P7OUT sets the data output to the port 7.

5:0 **Port 7 Output** 0: Output low
1: Output high

P8OUT : x'00FFC8'

Port 8 Output Register

8-bit access register

7	6	5	4	3	2	1	0
P8 OUT7	P8 OUT6	P8 OUT5	P8 OUT4	P8 OUT3	P8 OUT2	P8 OUT1	P8 OUT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

P8OUT sets the data output to the port 8.

7:0 Port 8 Output 0: Output low
1: Output high

P9OUT : x'00FFC9'

Port 9 Output Register

8-bit access register

7	6	5	4	3	2	1	0
P9 OUT7	P9 OUT6	P9 OUT5	P9 OUT4	P9 OUT3	P9 OUT2	P9 OUT1	P9 OUT0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

P9OUT sets the data output to the port 9.

7:0 Port 9 Output 0: Output low
1: Output high

PAOUT : x'00FFCA'

Port A Output Register

8-bit access register

7	6	5	4	3	2	1	0
-	-	PA OUT5	PA OUT4	PA OUT3	PA OUT2	PA OUT1	PA OUT0
R	R	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0/1	0/1	0/1	0/1	0/1	0/1

PAOUT sets the data output to the port A.

5:0 Port A Output 0: Output low
1: Output high

P

PBOUT : x'00FFCB'

Port B Output Register

8-bit access register

7	6	5	4	3	2	1	0
-	-	-	reserved	reserved	reserved	PB OUT1	PB OUT0
R	R	R	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0	0/1	0/1	0/1	0/1	0/1

PBOUT sets the data output to the port B.

4:2 Reserved Set to 0

1:0 Port B Output 0: Output low
1: Output high

P0IN : x'00FFD0'

Port 0 Input Register

8-bit access register

P0IN reads the port 0 data.

7	6	5	4	3	2	1	0
P0 IN7	P0 IN6	P0 IN5	P0 IN4	P0 IN3	P0 IN2	P0 IN1	P0 IN0
R	R	R	R	R	R	R	R
Port	Port	Port	Port	Port	Port	Port	Port
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 Port 0 Input 0: Input low
1: Input high

P1IN : x'00FFD1'

Port 1 Input Register

8-bit access register

P1IN reads the port 1 data.

7	6	5	4	3	2	1	0
P1 IN7	P1 IN6	P1 IN5	P1 IN4	P1 IN3	P1 IN2	P1 IN1	P1 IN0
R	R	R	R	R	R	R	R
Port	Port	Port	Port	Port	Port	Port	Port
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 Port 1 Input 0: Input low
1: Input high

P2IN : x'00FFD2'

Port 2 Input Register

8-bit access register

P2IN reads the port 2 data.

7	6	5	4	3	2	1	0
P2 IN7	P2 IN6	P2 IN5	P2 IN4	P2 IN3	P2 IN2	P2 IN1	P2 IN0
R	R	R	R	R	R	R	R
Port	Port	Port	Port	Port	Port	Port	Port
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 Port 2 Input 0: Input low
1: Input high

P3IN : x'00FFD3'

Port 3 Input Register

8-bit access register

P3IN reads the port 3 data.

7	6	5	4	3	2	1	0
P3 IN7	P3 IN6	P3 IN5	P3 IN4	P3 IN3	P3 IN2	P3 IN1	P3 IN0
R	R	R	R	R	R	R	R
Port	Port	Port	Port	Port	Port	Port	Port
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 Port 3 Input 0: Input low
1: Input high

P4IN : x'00FFD4'

Port 4 Input Register

8-bit access register

7	6	5	4	3	2	1	0
P4 IN7	P4 IN6	P4 IN5	P4 IN4	P4 IN3	P4 IN2	P4 IN1	P4 IN0
R	R	R	R	R	R	R	R
Port	Port	Port	Port	Port	Port	Port	Port
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 **Port 4 Input** 0: Input low
1: Input high

P4IN reads the port 4 data.

P5IN : x'00FFD5'

Port 5 Input Register

8-bit access register

7	6	5	4	3	2	1	0
P5 IN7	P5 IN6	P5 IN5	P5 IN4	P5 IN3	P5 IN2	P5 IN1	P5 IN0
R	R	R	R	R	R	R	R
Port	Port	Port	Port	Port	Port	Port	Port
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 **Port 5 Input** 0: Input low
1: Input high

P5IN reads the port 5 data.

P6IN : x'00FFD6'

Port 6 Input Register

8-bit access register

7	6	5	4	3	2	1	0
-	-	-	-	P6 IN3	P6 IN2	P6 IN1	P6 IN0
R	R	R	R	R	R	R	R
0	0	0	0	Port	Port	Port	Port
0	0	0	0	0/1	0/1	0/1	0/1

3:0 **Port 6 Input** 0: Input low
1: Input high

P6IN reads the port 6 data.

P

P7IN : x'00FFD7'

Port 7 Input Register

8-bit access register

7	6	5	4	3	2	1	0
-	P7 IN6	P7 IN5	P7 IN4	P7 IN3	P7 IN2	P7 IN1	P7 IN0
R	R	R	R	R	R	R	R
0	Port	Port	Port	Port	Port	Port	Port
0	0/1	0/1	0/1	0/1	0/1	0/1	0/1

6:0 **Port 7 Input** 0: Input low
1: Input high

P7IN reads the port 7 data. Reading P76 pin identifys the status input $\overline{\text{NMI}}$.

P8IN : x'00FFD8'

Port 8 Input Register

8-bit access register

P8IN reads the port 8 data.

7	6	5	4	3	2	1	0
P8 IN7	P8 IN6	P8 IN5	P8 IN4	P8 IN3	P8 IN2	P8 IN1	P8 IN0
R	R	R	R	R	R	R	R
Port	Port	Port	Port	Port	Port	Port	Port
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 Port 8 Input 0: Input low
1: Input high

P9IN : x'00FFD9'

Port 9 Input Register

8-bit access register

P9IN reads the port 9 data.

7	6	5	4	3	2	1	0
P9 IN7	P9 IN6	P9 IN5	P9 IN4	P9 IN3	P9 IN2	P9 IN1	P9 IN0
R	R	R	R	R	R	R	R
Port	Port	Port	Port	Port	Port	Port	Port
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 Port 9 Input 0: Input low
1: Input high

PAIN : x'00FFDA'

Port A Input Register

8-bit access register

PAIN reads the port A data.

7	6	5	4	3	2	1	0
-	-	PA IN5	PA IN4	PA IN3	PA IN2	PA IN1	PA IN0
R	R	R	R	R	R	R	R
0	0	Port	Port	Port	Port	Port	Port
0	0	0	0	0/1	0/1	0/1	0/1

5:0 Port A Input 0: Input low
1: Input high

PBIN : x'00FFDB'

Port B Input Register

8-bit access register

PBIN reads the port B data.

7	6	5	4	3	2	1	0
-	-	-	reserved	reserved	-	PB IN1	PB IN0
R	R	R	R	R	R	R	R
0	0	0	Port	Port	0	Port	Port
0	0	0	0/1	0/1	0	0/1	0/1

1:0 Port B Input 0: Input low
1: Input high

P0DIR : x'00FFE0'

Port 0 Input/Output Control Register

8-bit access register

P0DIR controls the port 0 input/ output.

7	6	5	4	3	2	1	0
P0 DIR7	P0 DIR6	P0 DIR5	P0 DIR4	P0 DIR3	P0 DIR2	P0 DIR1	P0 DIR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 **Port 0 Input/Output** 0: Input
1: Output

P1DIR : x'00FFE1'

Port 1 Input/Output Control Register

8-bit access register

P1DIR controls the port 1 input/ output.

7	6	5	4	3	2	1	0
P1 DIR7	P1 DIR6	P1 DIR5	P1 DIR4	P1 DIR3	P1 DIR2	P1 DIR1	P1 DIR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 **Port 1 Input/Output** 0: Input
1: Output

P2DIR : x'00FFE2'

Port 2 Input/Output Control Register

8-bit access register

P2DIR controls the port 2 input/ output.

7	6	5	4	3	2	1	0
P2 DIR7	P2 DIR6	P2 DIR5	P2 DIR4	P2 DIR3	P2 DIR2	P2 DIR1	P2 DIR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 **Port 2 Input/Output** 0: Input
1: Output

P3DIR : x'00FFE3'

Port 3 Input/Output Control Register

8-bit access register

P3DIR controls the port 3 input/ output.

7	6	5	4	3	2	1	0
P3 DIR7	P3 DIR6	P3 DIR5	P3 DIR4	P3 DIR3	P3 DIR2	P3 DIR1	P3 DIR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

7:0 **Port 3 Input/Output** 0: Input
1: Output

P4DIR : x'00FFE4'

Port 4 Input/Output Control Register

8-bit access register

7	6	5	4	3	2	1	0
P4 DIR7	P4 DIR6	P4 DIR5	P4 DIR4	P4 DIR3	P4 DIR2	P4 DIR1	P4 DIR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

P4DIR controls the port 4 input/output.

7:0 Port 4 Input/Output 0: Input
1: Output

P5DIR : x'00FFE5'

Port 5 Input/Output Control Register

8-bit access register

7	6	5	4	3	2	1	0
P5 DIR7	P5 DIR6	P5 DIR5	P5 DIR4	P5 DIR3	P5 DIR2	P5 DIR1	P5 DIR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

P5DIR controls the port 5 input/output.

7:0 Port 5 Input/Output 0: Input
1: Output

P6DIR : x'00FFE6'

Port 6 Input/Output Control Register

8-bit access register

7	6	5	4	3	2	1	0
-	-	-	-	P6 DIR3	P6 DIR2	P6 DIR1	P6 DIR0
R	R	R	R	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0/1	0/1	0/1	0/1

P6DIR controls the port 6 input/output.

3:0 Port 6 Input/Output 0: Input
1: Output

P7DIR : x'00FFE7'

Port 7 Input/Output Control Register

8-bit access register

7	6	5	4	3	2	1	0
-	-	P7 DIR5	P7 DIR4	P7 DIR3	P7 DIR2	P7 DIR1	P7 DIR0
R	R	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0/1	0/1	0/1	0/1	0/1	0/1

P7DIR controls the port 7 input/output.

5:0 Port 7 Input/Output 0: Input
1: Output

P8DIR : x'00FFE8'

Port 8 Input/Output Control Register

8-bit access register

7	6	5	4	3	2	1	0
P8 DIR7	P8 DIR6	P8 DIR5	P8 DIR4	P8 DIR3	P8 DIR2	P8 DIR1	P8 DIR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

P8DIR controls the port 8 input/ output.

7:0 Port 8 Input/Output 0: Input
1: Output

P9DIR : x'00FFE9'

Port 9 Input/Output Control Register

8-bit access register

7	6	5	4	3	2	1	0
P9 DIR7	P9 DIR6	P9 DIR5	P9 DIR4	P9 DIR3	P9 DIR2	P9 DIR1	P9 DIR0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

P9DIR controls the port 9 input/ output.

7:0 Port 9 Input/Output 0: Input
1: Output

PADIR : x'00FFEA'

Port A Input/Output Control Register

8-bit access register

7	6	5	4	3	2	1	0
-	-	PA DIR5	PA DIR4	PA DIR3	PA DIR2	PA DIR1	PA DIR0
R	R	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0/1	0/1	0/1	0/1	0/1	0/1

PADIR controls the port A input/ output.

5:0 Port A Input/Output 0: Input
1: Output

P

PBDIR : x'00FFEB'

Port B Input/Output Control Register

8-bit access register

7	6	5	4	3	2	1	0
-	-	-	reserv ed	reserv ed	reserv ed	PB DIR1	PB DIR0
R	R	R	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0	0/1	0/1	0/1	0/1	0/1

PBDIR controls the port B input/ output.

4-2 Reserved Set to 0.

1-0 Port B Input/Output 0: Input
1: Output

	7	6	5	4	3	2	1	0
	-	-	-	-	-	-	P0 MD1	P0 MD0
	R	R	R	R	R	R	R/W	R/W
Processor address/data separate mode	0	0	0	0	0	0	0	1
Processor address/data shared mode	0	0	0	0	0	0	1	0
Other modes	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0/1	0/1

P0MD : x'00FFF0'

Port 0 Mode Register

8-bit access register

1:0 P0 Input/Output Signal Switch

00: Port
01: Data
10: Address/data shared mode

P0MD sets a signal output to the port 0.

	7	6	5	4	3	2	1	0
	-	P1 LMD6	P1 LMD5	P1 LMD4	P1 LMD3	P1 LMD2	P1 LMD1	P1 LMD0
	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Processor address/data separate mode (16-bit bus width)	0	0	0	0	0	0	0	1
Processor address/data shread mode (8-bit bus width)	0	0	0	0	0	0	1	0
Other modes	0	0	0	0	0	0	0	0
	0	0/1	0/1	0/1	0/1	0/1	0/1	0/1

P1LMD : x'00FFF2'

Port 1 Mode Register L

8-bit access register

P1LMD sets a signal output to the port 1.

6:5 P12 Input/Output Signal Switch

00: Port
01: TM11IOA input
10: TM11IOA output

4 P11 Input/Output Signal Switch

0: Port
1: Reserved

3:2 P10 Input/Output Signal Switch

00: Port
01: TM8IOB input
10: TM8IOB output

1:0 P1 Input/Output Signal Switch

00: Port/each funtion
01: Data output
10: Address/data shared mode

When P1 is used as a port or an input/output pin of each peripheral function, always set P1LMD[1:0] to 00.

7	6	5	4	3	2	1	0
P1 HMD7	P1 HMD6	P1 HMD5	P1 HMD4	P1 HMD3	P1 HMD2	P1 HMD1	P1 HMD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

P1HMD : x'00FFF3'

Port 1 Mode Register H

8-bit access register

P1HMD sets a signal output to the port 1.

7 **P17 Input/Output Signal Switch**
0: Port
1: TM12IC input

6:5 **P16 Input/Output Signal Switch**
00: Port
01: TM12IOB input
10: TM12IOB output

4:3 **P15 Input/Output Signal Switch**
00: Port
01: TM12IOA input
10: TM12IOA output

2 **P14 Input/Output Signal Switch**
0: Port
1: TM11IC input

1:0 **P13 Input/Output Signal Switch**
00: Port
01: TM11IOB input
10: TM11IOB output

When P1 is used as a port or an input/output pin of each peripheral function, always set P1LMD[1:0] to 00.

P2MD : x'00FFF1'

Port 2 Mode Register

8-bit access register

	7	6	5	4	3	2	1	0
	P2 MD7	-	P2 MD5	P2 MD4	P2 MD3	P2 MD2	-	P2 MD0
	R/W	R	R/W	R/W	R/W	R/W	R	R/W
Processor address/data separate mode	0	0	0	0	0	0	0	1
Other modes	0	0	0	0	0	0	0	0
	0/1	0	0/1	0/1	0/1	0/1	0	0/1

P2MD sets a signal output to the port 2.

7 P24 Input/Output Signal Switch
0: Port
1: TM15IA input
(cannot use P56 as TM15IA input)

5 P22 Input/Output Signal Switch
0: Port
1: SBO2 output

4 P21 Input/Output Signal Switch
0: Port
1: SBI2 input
(cannot use P82 as SBI2 input)

3:2 P20 Input/Output Signal Switch
00: Port
01: SBT2 input
(cannot use P60 as SBT2 input)
10: SBT2 half-duplex output
11: SBT2 output

0 P2 Input/Output Signal Switch
0: Port/each Function
1: Address

P23, P25, P26 and P27 can be used as ports if P2MD0 is set to 0. When P2 is uesed as a port or an input/output pin of each peripheral function, P2MD0 is always set to 0.

P

P3LMD : x'00FFF4'

Port 3 Mode Register L

8-bit access register

	7	6	5	4	3	2	1	0
	P3 LMD7	P3 LMD6	P3 LMD5	P3 LMD4	P3 LMD3	P3 LMD2	P3 LMD1	P3 LMD0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Processor address/data separate mode	0	1	0	1	0	1	0	1
Other modes	0	0	0	0	0	0	0	0
	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

P3LMD sets a signal output to the port 3.

7:6 P33 Input/Output Signal Switch

00: Port
01: A11 output
10: $\overline{\text{KI3}}$ input

5:4 P32 Input/Output Signal Switch

00: Port
01: A10 output
10: $\overline{\text{KI2}}$ input

3:2 P31 Input/Output Signal Switch

00: Port
01: A9 output
10: $\overline{\text{KI1}}$ input

1:0 P30 Input/Output Signal Switch

00: Port
01: A8 output
10: $\overline{\text{KI0}}$ input

P3HMD : x'00FFF5'

Port 3 Mode Register H

8-bit access register

	7	6	5	4	3	2	1	0
	P3 HMD7	P3 HMD6	P3 HMD5	P3 HMD4	P3 HMD3	P3 HMD2	P3 HMD1	P3 HMD0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Processor address/data separate mode	0	1	0	1	0	1	0	1
Other modes	0	0	0	0	0	0	0	0
	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

P3HMD sets a signal output to the port 3.

7:6 P37 Input/Output Signal Switch
00: Port
01: A15 output
10: $\overline{\text{KI7}}$ input

5:4 P36 Input/Output Signal Switch
00: Port
01: A14 output
10: $\overline{\text{KI6}}$ input

3:2 P35 Input/Output Signal Switch
00: Port
01: A13 output
10: $\overline{\text{KI5}}$ input

1:0 P34 Input/Output Signal Switch
00: Port
01: A12 output
10: $\overline{\text{KI4}}$ input

P

	7	6	5	4	3	2	1	0
	P4 LMD7	P4 LMD6	P4 LMD5	P4 LMD4	P4 LMD3	P4 LMD2	P4 LMD1	P4 LMD0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Processor mode	0	1	0	1	1	1	1	1
Other modes	0	0	0	0	0	0	0	0
	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

P4LMD : x'00FFF6'

Port 4 Mode Register L

8-bit access register

P4LMD sets a signal output to the port 4.

7:6 P45 Input/Output Signal Switch
00: Port
01: A21 output
10: AN5

5:4 P44 Input/Output Signal Switch
00: Port
01: A20 output
10: AN4

3 P43 Input/Output Signal Switch
0: Port
1: A19 output

2 P42 Input/Output Signal Switch
0: Port
1: A18 output

1 P41 Input/Output Signal Switch
0: Port
1: A17 output

0 P40 Input/Output Signal Switch
0: Port
1: A16 output

P4HMD : x'00FFF7'

Port 4 Mode Register H

8-bit access register

P4HMD sets a signal output to the port 4.

	7	6	5	4	3	2	1	0
	-	-	-	-	P4 HMD3	P4 HMD2	P4 HMD1	P4 HMD0
	R	R	R	R	R/W	R/W	R/W	R/W
Processor mode	0	0	0	0	0	1	0	1
Other modes	0	0	0	0	0	0	0	0
	0	0	0	0	0/1	0/1	0/1	0/1

3:2 P47 Input/Output Signal Switch

00: Port
01: A23 output
10: AN7
11: WDOUT output

1:0 P46 Input/Output Signal Switch

00: Port
01: A22 output
10: AN6
11: STOP output

P

P5LMD : x'00FFF8'

Port 5 Mode Register L

8-bit access register

	7	6	5	4	3	2	1	0
	P5 LMD7	P5 LMD6	P5 LMD5	P5 LMD4	P5 LMD3	P5 LMD2	P5 LMD1	P5 LMD0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Processor mode	0	1	0	1	0	1	0	1
Other modes	0	0	0	0	0	0	0	0
	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

P5LMD sets a signal output to the port 5.

7:6 P53 Input/Output Signal Switch

00: Port
01: $\overline{CS3}$ output
10: TM14OB output

5:4 P52 Input/Output Signal Switch

00: Port
01: $\overline{CS2}$ output
10: TM14OA output

3:2 P51 Input/Output Signal Switch

00: Port
01: $\overline{CS1}$ output
10: TM13OB output

1:0 P50 Input/Output Signal Switch

00: Port
01: $\overline{CS0}$ output
10: TM13OA output

	7	6	5	4	3	2	1	0
	-	-	-	P5 HMD4	P5 HMD3	P5 HMD2	P5 HMD1	P5 HMD0
	R	R	R	R/W	R/W	R/W	R/W	R/W
Processor address/data separate mode	0	0	0	0	1	1	0	0
Processor address/data shared mode	0	0	0	0	0	1	0	0
Other modes	0	0	0	0	0	0	0	0
	0	0	0	0/1	0/1	0/1	0/1	0/1

P5HMD : x'00FFF9'

Port 5 Mode Register H

8-bit access register

P5HMD sets a signal output to the port 5.

4:2 P56 Input/Output Signal Switch

000: Port
001: ALE output
010: $\overline{\text{ALE}}$ output
011: $\overline{\text{BSTRE}}$ output
100: TM15IA
(cannot use P24 as TM15IA input)

1 P55 Input/Output Signal Switch

0: Port
1: $\overline{\text{BRACK}}$ output

0 P54 Input/Output Signal Switch

0: Port
1: $\overline{\text{BREQ}}$ input

P

P6MD : x'00FFFF'

Port 6 Mode Register

8-bit access register

	7	6	5	4	3	2	1	0
	-	-	P6 MD5	P6 MD4	P6 MD3	P6 MD2	P6 MD1	P6 MD0
	R	R	R/W	R/W	R/W	R/W	R/W	R/W
Processor mode (16-bit)	0	0	1	1	1	0	0	0
Processor mode (8-bit)	0	0	0	1	1	0	0	0
Other modes	0	0	0	0	0	0	0	0
	0	0	0/1	0/1	0/1	0/1	0/1	0/1

P6MD sets a signal output to the port 6.

5 P63 Input/Output Signal Switch
0: Port
1: $\overline{\text{WEH}}$ ($\overline{\text{WE}}$) output

4 P62 Input/Output Signal Switch
0: Port
1: $\overline{\text{WEL}}$ output

3 P61 Input/Output Signal Switch
0: Port
1: $\overline{\text{RE}}$ output

2:0 P60 Input/Output Signal Switch
000: Port
001: WAIT input
010: SBT2 input
(cannot use P20 as SBT2 input)
011: SBT2 output
100: SBT2 half-duplex output

7	6	5	4	3	2	1	0
-	P7 LMD6	P7 LMD5	P7 LMD4	P7 LMD3	P7 LMD2	P7 LMD1	P7 LMD0
R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0/1	0/1	0/1	0/1	0/1	0/1	0/1

P7LMD : x'00FFFA'

Port 7 Mode Register L

8-bit access register

P7LMD sets a signal output to the port 7.

6:5 P72 Input/Output Signal Switch

00: Port
01: SBO0 output
10: $\overline{\text{UCAS}}$ output

4:3 P71 Input/Output Signal Switch

00: Port
01: SBI0 input
10: $\overline{\text{LCAS}}$ output
11: $\overline{\text{CAS}}$ output

2:0 P70 Input/Output Signal Switch

000: Port
001: SBT0 input
010: SBT0 output
011: SBT0 half-duplex output
101: $\overline{\text{RAS}}$ output

P

7	6	5	4	3	2	1	0
-	P7 HMD6	P7 HMD5	P7 HMD4	P7 HMD3	P7 HMD2	P7 HMD1	P7 HMD0
R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0/1	0/1	0/1	0/1	0/1	0/1	0/1

P7HMD : x'00FFFB'

Port 7 Mode Register H

8-bit access register

P7HMD sets a signal output to the port 7.

6:5 **P75 Input/Output Signal Switch**
00: Port
01: SBO1 output

4:3 **P74 Input/Output Signal Switch**
00: Port
01: SBI1 input

2:0 **P73 Input/Output Signal Switch**
000: Port
001: SBT1 input
010: SBT1 output
011: SBT1 half-duplex output
100: $\overline{\text{DMUX}}$ output

7	6	5	4	3	2	1	0
-	-	-	P8 LMD4	P8 LMD3	P8 LMD2	P8 LMD1	P8 LMD0
R	R	R	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0	0/1	0/1	0/1	0/1	0/1

P8LMD : x'00FFFC'

Port 8 Mode Register L

8-bit access register

P8LMD sets a signal output to the port 8.

4:2 P82 Input/Output Signal Switch

000: Port
001: TM0IO input, SBT3 input
010: TM0IO output
011: SBT3 output
100: SBT3 half-duplex output
101: SCL3 open drain output
110: SBT2 input
(cannot use P21 as SBI2 input)

1 P81 Input/Output Signal Switch

0: Port
1: DAC1 output

0 P80 Input/Output Signal Switch

0: Port
1: DAC0 output

7	6	5	4	3	2	1	0
P8 MMD7	P8 MMD6	P8 MMD5	P8 MMD4	P8 MMD3	P8 MMD2	P8 MMD1	P8 MMD0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1

P8MMD : x'00FFFD'

Port 8 Mode Register M

8-bit access register

P8MMD sets a signal output to the port 8.

7:5 P85 Input/Output Signal Switch

000: Port
001: TM9IOA input, SBT4 input
010: TM9IOA output
011: SBT4 output
100: SBT4 half-duplex output
101: SCL4 open drain output
110: SBO2 output

4:2 P84 Input/Output Signal Switch

000: Port
001: TM7IO input
010: TM7IO output
011: SBO3 output
100: SBD3 open drain input/output

1:0 P83 Input/Output Signal Switch

00: Port
01: TM4IO input, SBI3 input
10: TM4IO output
11: Reserved

7	6	5	4	3	2	1	0
-	-	-	-	P8 HMD3	P8 HMD2	P8 HMD1	P8 HMD0
R	R	R	R	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0	0	0/1	0/1	0/1	0/1

P8HMD : x'00FFFE'

Port 8 Mode Register H

8-bit access register

P8HMD sets a signal output to the port 8.

3:2 P87 Input/Output Signal Switch

00: Port
01: SBO4 output
10: SBD4 open drain output, SBD4 input
11: TM9IC input

1:0 P86 Input/Output Signal Switch

00: Port
01: TM9IOB input, SBI4 input
10: TM9IOB output

P

7	6	5	4	3	2	1	0
P9 LMD7	P9 LMD6	P9 LMD5	P9 LMD4	P9 LMD3	-	P9 LMD1	P9 LMD0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W
0	0	0	0	0	0	0	0
0/1	0/1	0/1	0/1	0/1	0	0/1	0/1

P9LMD : x'00FFEC'

Port 9 Mode Register L

8-bit access register

P9LMD sets a signal output to the port 9.

7:6 P92 Input/Output Signal Switch

00: Port
01: TM10IOB input, $\overline{\text{DMAREQ0}}$ input
10: TM10IOB output

5:3 P91 Input/Output Signal Switch

000: Port
001: TM10IOA input
010: TM10IOA output
011: BIBT2 output
100: $\overline{\text{DMAACK1}}$ output

1:0 P90 Input/Output Signal Switch

00: Port
01: TM8IOA input, $\overline{\text{DMAREQ1}}$ input
10: TM8IOA output
11: BIBT1 output

7	6	5	4	3	2	1	0
-	-	P9 HMD5	P9 HMD4	P9 HMD3	P9 HMD2	P9 HMD1	P9 HMD0
R	R	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0/1	0/1	0/1	0/1	0/1	0/1

P9HMD : x'00FFED'

Port 9 Module Register H

8-bit access register

P9HMD sets a signal output to the port 9.

5	**P97 Input/Output Signal Switch**	0: Port 1: AN3 input
4	**P96 Input/Output Signal Switch**	0: Port 1: AN2 input
3	**P95 Input/Output Signal Switch**	0: Port 1: AN1 input
2	**P94 Input/Output Signal Switch**	0: Port 1: AN0 input
1:0	**P93 Input/Output Signal Switch**	00: Port 01: TM10IC input 10: $\overline{\text{DMAACK0}}$ output

P

PAMD : x'00FFDC'

Port A Mode Register

8-bit access register

7	6	5	4	3	2	1	0
-	-	-	PA MD4	PA MD3	PA MD2	PA MD1	PA MD0
R	R	R	R/W	R/W	R/W	R/W	R/W
0	0	0	0	0	0	0	0
0	0	0	0/1	0/1	0/1	0/1	0/1

PAMD sets a signal output to the port A.

4	**PA4 Input/Output Signal Switch**	0: Port 1: $\overline{\text{IRQ4}}$ input or TM15IB input
3	**PA3 Input/Output Signal Switch**	0: Port 1: $\overline{\text{IRQ3}}$ input
2	**PA2 Input/Output Signal Switch**	0: Port 1: $\overline{\text{IRQ2}}$ input
1	**PA1 Input/Output Signal Switch**	0: Port 1: $\overline{\text{IRQ1}}$ input
0	**PA0 Input/Output Signal Switch**	0: Port 1: $\overline{\text{IRQ0}}$ input

PBMD : x'00FFDD'

Port B Mode Register

8-bit access register

7	6	5	4	3	2	1	0
-	reserved	reserved	reserved	reserved	PB MD2	PB MD1	PB MD0
R	R/W	R/W	R/W	R/W	R/W	R/W	R/W
0	0	0	0	1	1	0	1
0	0/1	0/1	0/1	0/1	0/1	0/1	0/1

PBMD sets a signal output to the port B.

6:4	**Reserved**	Set to 0
3	**Reserved**	Set to 1
2	**PB1 Input/Output Signal Switch**	0: Port 1: XI input
1:0	**PB0 Input/Output Signal Switch**	00: Port 01: BOSC output 10: BIBT1 output 11: BIBT2 output

P

11-2-2 MN102H55D/55G/F55G Address Map

Upper 20 bits \ Lower 4 bits	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	
x'00FC00'	IAGR		reserved		reserved		EFCR		reserved		reserved		reserved		CPUM		Internal Control Registers
x'00FC40'					reserved		reserved		reserved	❍ EIICR	reserved	❍ UNICR	reserved	❍ WDICR	reserved	❍ NMICR	Interrupt Control Registers
x'00FC50'	❍ TM9UICH	❍ TM9UICL	❍ TM8BICH	❍ TM8BICL	❍ TM1UICH	❍ TM1UICL	❍ IQ1ICH	❍ IQ1ICL	❍ TM8AICH	❍ TM8AICL	❍ TM8UICH	❍ TM8UICL	❍ TM0UICH	❍ TM0UICL	❍ IQ0ICH	❍ IQ0ICL	
x'00FC60'	❍ TM10AICH	❍ TM10AICL	❍ TM10UICH	❍ TM10UICL	❍ TM3UICH	❍ TM3UICL	❍ IQ3ICH	❍ IQ3ICL	❍ TM9BICH	❍ TM9BICL	❍ TM9AICH	❍ TM9AICL	❍ TM2UICH	❍ TM2UICL	❍ IQ2ICH	❍ IQ2ICL	
x'00FC70'	❍ TM11BICH	❍ TM11BICL	❍ TM11AICH	❍ TM11AICL	❍ TM5UICH	❍ TM5UICL	❍ KIICH	❍ KIICL	❍ TM11UICH	❍ TM11UICL	❍ TM10BICH	❍ TM10BICL	❍ TM4UICH	❍ TM4UICL	❍ IQ4ICH	❍ IQ4ICL	
x'00FC80'					❍ TM12BICH	❍ TM12BICL	❍ TM7UICH	❍ TM7UICL	❍ TM12AICH	❍ TM12AICL	❍ TM12UICH	❍ TM12UICL	❍ TM6UICH	❍ TM6UICL	❍ ADICH	❍ ADICL	
x'00FC90'	❍ SC3RICH	❍ SC3RICL	❍ SC3TICH	❍ SC3TICL	❍ SC2RICH	❍ SC2RICL	❍ SC2TICH	❍ SC2TICL	❍ SC1RICH	❍ SC1RICL	❍ SC1TICH	❍ SC1TICL	❍ SC0RICH	❍ SC0RICL	❍ SC0TICH	❍ SC0TICL	
x'00FCA0'	❍ AT3ICH	❍ AT3ICL	❍ AT2ICH	❍ AT2ICL	❍ AT1ICH	❍ AT1ICL	❍ AT0ICH	❍ AT0ICL	❍ ETC1ICH	❍ ETC1ICL	❍ ETC0ICH	❍ ETC0ICL	❍ SC4RICH	❍ SC4RICL	❍ SC4TICH	❍ SC4TICL	
x'00FCB0'									WDREG		KEYCTR		KEYTRG		IRQTRG		
x'00FCD0'						❍ ADBCTL		ADB1H	ADB1L ✻			ADB0H	ADB0L ✻			❍ SYSCTL	System Control
x'00FD00'						AT0DSTH	AT0DSTL ✻			AT0SRCH	AT0SRCL ✻		AT0CNT ◆		AT0CTR ◆		ATC
x'00FD10'						AT1DSTH	AT1DSTL ✻			AT1SRCH	AT1SRCL ✻		AT1CNT ◆		AT1CTR ◆		
x'00FD20'						AT2DSTH	AT2DSTL ✻			AT2SRCH	AT2SRCL ✻		AT2CNT ◆		AT2CTR ◆		
x'00FD30'						AT3DSTH	AT3DSTL ✻			AT3SRCH	AT3SRCL ✻		AT3CNT ◆		AT3CTR ◆		
x'00FD40'						ET0DSTH	ET0DSTL ✻			ET0SRCH	ET0SRCL ✻		ET0CNT ◆		ET0CTR ◆		ETC
x'00FD50'						ET1DSTH	ET1DSTL ✻			ET1SRCH	ET1SRCL ✻		ET1CNT ◆		ET1CTR ◆		
x'00FD80'					❍ SC1STR	❍ SC1TRB	SC1CTR						❍ SC0STR	❍ SC0TRB	SC0CTR		Serial Interface
x'00FD90'					❍ SC3STR	❍ SC3TRB	SC3CTR						❍ SC2STR	❍ SC2TRB	SC2CTR		
x'00FDA0'					reserved	reserved	reserved						❍ SC4STR	❍ SC4TRB	SC4CTR		

❍ : 8-bit access Use the MOVB instruction.

◆ : 16-bit access during write operation Use the MOV instruction.

✻ : 16-bit or 24-bit access during write operation Use the MOV instruction or the MOVX instruction.

❑ : This register cannot neither read nor write. This register is used as a double buffer of the compare register when the PWM functions is selected.

Upper 20 bits \ Lower 4 bits	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	
x'00FE00'					❍ TM14CA	❍ TM13CA	❍ TM14BC	❍ TM13BC	❍ TM7BC	❍ TM6BC	❍ TM5BC	❍ TM4BC	❍ TM3BC	❍ TM2BC	❍ TM1BC	❍ TM0BC	8-bit Timers
x'00FE10'					❍ TM14CB	❍ TM13CB	❍ TM14BR	❍ TM13BR	❍ TM7BR	❍ TM6BR	❍ TM5BR	❍ TM4BR	❍ TM3BR	❍ TM2BR	❍ TM1BR	❍ TM0BR	
x'00FE20'							❍ TM14MD	❍ TM13MD	❍ TM7MD	❍ TM6MD	❍ TM5MD	❍ TM4MD	❍ TM3MD	❍ TM2MD	❍ TM1MD	❍ TM0MD	
x'00FE30'																	16-bit Timers
x'00FE80'		❍ TM8MD2	reserved		TM8CBX ❑		TM8CB		TM8CAX ❑		TM8CA		TM8BC		TM8MD		
x'00FE90'		❍ TM9MD2	reserved		TM9CBX ❑		TM9CB		TM9CAX ❑		TM9CA		TM9BC		TM9MD		
x'00FEA0'		❍ TM10MD2	reserved		TM10CBX ❑		TM10CB		TM10CAX ❑		TM10CA		TM10BC		TM10MD		
x'00FEB0'			reserved		TM11CBX ❑		TM11CB		TM11CAX ❑		TM11CA		TM11BC		TM11MD		
x'00FEC0'			reserved		TM12CBX ❑		TM12CB		TM12CAX ❑		TM12CA		TM12BC		TM12MD		
x'00FED0'											TM15CA		TM15BC		TM15MD		
x'00FF00'	AN3BUF		AN2BUF		AN1BUF		AN0BUF				reserved				ANCTR		AD Converter
x'00FF10'									AN7BUF		AN6BUF		AN5BUF		AN4BUF		
x'00FF40'													❍ DA1BUF	❍ DA1CTR	❍ DA0BUF	❍ DA0CTR	DA Converter
x'00FF70'	reserved		reserved		reserved		reserved		reserved		reserved		reserved		reserved		
x'00FF80'			MPXADR ◆		ALEEDGE ◆		WEEDGE ◆		REEDGE ◆		MEMMD2 ◆		MEMMD1 ◆		EXWMD ◆		CPU Memory Control
x'00FF90'	reserved		reserved										DRAMMD2 ◆		DRAMMD1 ◆		
x'00FFB0'		PBSWH	PBSWL		❍ PBPLU	❍ PAPLU	❍ P9PLU	❍ P8PLU	❍ P7PLU	❍ P6PLU	❍ P5PLU	❍ P4PLU	❍ P3PLU	❍ P2PLU	❍ P1PLU	❍ P0PLU	Port Control
x'00FFC0'	LBSWH		LBSWL		❍ PBOUT	❍ PAOUT	❍ P9OUT	❍ P8OUT	❍ P7OUT	❍ P6OUT	❍ P5OUT	❍ P4OUT	❍ P3OUT	❍ P2OUT	❍ P1OUT	❍ P0OUT	
x'00FFD0'			❍ PBMD	❍ PAMD	❍ PBIN	❍ PAIN	❍ P9IN	❍ P8IN	❍ P7IN	❍ P6IN	❍ P5IN	❍ P4IN	❍ P3IN	❍ P2IN	❍ P1IN	❍ P0IN	
x'00FFE0'			❍ P9HMD	❍ P9LMD	❍ PBDIR	❍ PADIR	❍ P9DIR	❍ P8DIR	❍ P7DIR	❍ P6DIR	❍ P5DIR	❍ P4DIR	❍ P3DIR	❍ P2DIR	❍ P1DIR	❍ P0DIR	
x'00FFF0'	❍ P6MD	❍ P8HMD	❍ P8MMD	❍ P8LMD	❍ P7HMD	❍ P7LMD	❍ P5HMD	❍ P5LMD	❍ P4HMD	❍ P4LMD	❍ P3HMD	❍ P3LMD	❍ P1HMD	❍ P1LMD	❍ P2MD	❍ P0MD	

❍ : 8-bit access Use the MOVB instruction.

◆ : 16-bit access during write operation Use the MOV instruction.

✻ : 16-bit or 24-bit access during write operation Use the MOV instruction or the MOVX instruction.

❑ : This register cannot neither read nor write. This register is used as a double buffer of the compare register when the PWM functions is selected.

11-2-3 List of Pin Functions

EO = External Oscillation

	Pin Name	Input Level	Output Level	Schumitt	Pull-up	RESET Note 1	RESET Note 2	RESET Note 3	BREQ="L"	STOP/HALT
1	P60,WAIT	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
2	P61,/RE	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z at /RE	Hi-Z at /RE
3	P62,/WEL	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z at /WEL	Hi-Z at /WEL
4	P63,/WEH	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z at /WEH	Hi-Z at /WEH
5	P50,/CS0	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z at /CS0	Hi-Z at /CS0
6	P51,/CS1	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z at /CS1	Hi-Z at /CS1
7	P52,/CS2	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z at /CS2	Hi-Z at /CS2
8	P53,/CS3	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z at /CS3	Hi-Z at /CS3
9	P54,/BREQ	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Low	✻
10	P55,/BRACK	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Low	✻
11	P56,ALE,/ALE,/BSTRE	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Low	Hi-Z except P56	Hi-Z except P56
12	P57,/WORD	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
13	P20,A00	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Hi-Z	Hi-Z at A00	Hi-Z at A00
14	P21,A01	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Hi-Z	Hi-Z at A01	Hi-Z at A01
15	P22,A02	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Hi-Z	Hi-Z at A02	Hi-Z at A02
16	P23,A03	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Hi-Z	Hi-Z at A03	Hi-Z at A03
17	VDD	-	-	-	-	-	-	-	-	-
18	BOSC	TTL	CMOS	Yes	Programmable	Low	Low	Low	✻	Note 4
19	VSS	-	-	-	-	-	-	-	-	-
20	XI	CMOS	-	-	-	-	-	-	-	-
21	XO	-	-	-	-	High (EO)	High (EO)	High (EO)	✻	Note 5
22	VDD	-	-	-	-	-	-	-	-	-
23	OSCI	CMOS	-	-	-	-	-	-	-	-
24	OSCO	-	-	-	-	High (EO)	High (EO)	High (EO)	✻	Note 6
25	MODE	CMOS	-	Yes	No	High (Input)	High (Input)	High (Input)	MODE	MODE
26	P24,A04	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Hi-Z	Hi-Z at A04	Hi-Z at A04
27	P25,A05	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Hi-Z	Hi-Z at A05	Hi-Z at A05
28	P26,A06	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Hi-Z	Hi-Z at A06	Hi-Z at A06
29	P27,A07	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Hi-Z	Hi-Z at A07	Hi-Z at A07
30	P30,A08	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Hi-Z	Hi-Z at A08	Hi-Z at A08
31	P31,A09	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Hi-Z	Hi-Z at A09	Hi-Z at A09
32	P32,A10	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Hi-Z	Hi-Z at A10	Hi-Z at A10
33	P33,A11	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Hi-Z	Hi-Z at A11	Hi-Z at A11
34	AVDD	-	-	-	-	-	-	-	-	-
35	P34,A12	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Hi-Z	Hi-Z at A12	Hi-Z at A12
36	P35,A13	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Hi-Z	Hi-Z at A13	Hi-Z at A13
37	P36,A14	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Hi-Z	Hi-Z at A14	Hi-Z at A14
38	P37,A15	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Hi-Z	Hi-Z at A15	Hi-Z at A15
39	P40,A16	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Undefined	Hi-Z at A16	Hi-Z at A16
40	P41,A17	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Undefined	Hi-Z at A17	Hi-Z at A17
41	P42,A18	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Undefined	Hi-Z at A18	Hi-Z at A18
42	P43,A19	TTL	CMOS	Yes	Programmable	Hi-Z	Undefined	Undefined	Hi-Z at A19	Hi-Z at A19
43	Vref-	-	-	-	-	-	-	-	-	-
44	P44,A20,AN4	Analog,CMOS	CMOS	No	Programmable	Hi-Z	Undefined	Undefined	Hi-Z at A20	Hi-Z at A20
45	P45,A21,AN5	Analog,CMOS	CMOS	No	Programmable	Hi-Z	Undefined	Undefined	Hi-Z at A21	Hi-Z at A21
46	P46,A22,STOP,AN6	Analog,CMOS	CMOS	No	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z at A22	Hi-Z at A22
47	P47,A23,WDOUT,AN7	Analog,CMOS	CMOS	No	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z at A23	Hi-Z at A23
48	P80,DAC0	Analog,CMOS	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
49	P81,DAC1	Analog,CMOS	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
50	P82,TM2IO	Analog,CMOS	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻

51	P83,TM3IO	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
52	P84,TM4IO	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
53	P85,TM5IO	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
54	Vref+	-	-	-	-	-	-	-	-	-
55	P86,TM6IOA	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
56	P87,TM6IOB	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
57	P90,TM6IC	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
58	P91,TM7IOA	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
59	P92,TM7IOB	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
60	P93,TM7IC	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
61	AVSS	-	-	-	-	-	-	-	-	-
62	P94,AN0	Analog,CMOS	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
63	P95,AN1	Analog,CMOS	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
64	P96,AN2	Analog,CMOS	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
65	P97,AN3	Analog,CMOS	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
66	VDD(VPP)	-	-	-	-	-	-	-	-	-
67	P70,SBT0	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
68	P71,SBI0	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
69	P72,SBO0	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
70	P73,SBT1	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
71	P74,SBI1	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
72	P75,SBO1	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
73	Pull-up	-	-	-	-	-	-	-	-	-
74	Pull-up	-	-	-	-	-	-	-	-	-
75	/NMI	CMOS	-	Yes	No	/NMI	/NMI	/NMI	/NMI	/NMI
76	PA0,/IRQ0	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	*	✻
77	PA1,/IRQ1	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
78	PA2,/IRQ2	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
79	PA3,/IRQ3	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
80	PA4,/IRQ4	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	✻	✻
81	PA5,ADSEP	TTL	CMOS	Yes	Programmable	Hi-Z	High (Input)	Low (Input)	✻	✻
82	/RST	CMOS	-	Yes	No	Low (Input)	Low (Input)	Low (Input)	High	High
83	VDD	-	-	-	-	-	-	-	-	-
84	P00,D00,AD00	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z excapt P00	Hi-Z excapt P00
85	P01,D01,AD01	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z excapt P01	Hi-Z excapt P01
86	P02,D02,AD02	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z excapt P02	Hi-Z excapt P02
87	P03,D03,AD03	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z excapt P03	Hi-Z excapt P03
88	P04,D04,AD04	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z excapt P04	Hi-Z excapt P04
89	P05,D05,AD05	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z excapt P05	Hi-Z excapt P05
90	P06,D06,AD06	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z excapt P06	Hi-Z excapt P06
91	P07,D07,AD07	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z excapt P07	Hi-Z excapt P07
92	VSS	-	-	-	-	-	-	-	-	-
93	P10,D08,AD08	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z excapt P10	Hi-Z excapt P10
94	P11,D09,AD09	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z excapt P11	Hi-Z excapt P11
95	P12,D10,AD10	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z excapt P12	Hi-Z excapt P12
96	P13,D11,AD11	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z excapt P13	Hi-Z excapt P13
97	P14,D12,AD12	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z excapt P14	Hi-Z excapt P14
98	P15,D13,AD13	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z excapt P15	Hi-Z excapt P15
99	P16,D14,AD14	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z excapt P16	Hi-Z excapt P16
100	P17,D15,AD15	TTL	CMOS	Yes	Programmable	Hi-Z	Hi-Z	Hi-Z	Hi-Z excapt P17	Hi-Z excapt P17

✻ : Depends on pin setting   Note 1: Single-chip mode   Note 2: Processor mode (address/data separated mode)
Note 3: Processor mode (address/data shared mode)   Note 4: Low during STOP0/1 mode
Note 5: High during STOP0/1 mode   Note 6: High during STOP0/1, HALT1 mode

11-3 Initialization Program

The initialization program must be executed first after reset release. The initialization program should be allocated on x'80000' in single-chip mode, memory expansion mode or processor mode.

```
; Initialization Program
start   equ     *       ; x'080000'
        jmp     init

init    equ     *

; Register Initialization
        sub     d0,d0
        mov     d0,d1
        mov     d0,d2
        mov     d0,d3
        mov     d0,a0
        mov     d0,a1
        mov     d0,a2

        mov     STACK_TOP,a3
        mov     d0,mdr

; Memory Mode Setting
        mov     EXW_INT,d0
        mov     d0,(Exwmd)
        mov     MEM1_INIT,d0
        mov     d0,(Memmd1)
        mov     MEM2_INIT,d0
        mov     d0,(Memmd2)
        mov     DRAM1_INIT,d0
        mov     d0,(Drammd1)
        mov     DRAM2_INIT,d0
        mov     d0,(Drammd2)

; Other Setting
        ...........

; Interrupt Enable
        mov     INIT_PSW,d0
        mov     d0,psw
```

Clear register to 0. Execute this operation although this step is not always required.

Set the initial value of the stack pointer. (Always set the even address.)

Set the number of waits for external memory space.

Select the external memory bus width.

Set the mode such as burst ROM mode.

Set the DRAM mode.

Recommend to initialize the port in this step.

11-4 Flash EEPROM Version

11-4-1 Overview

The MN102HF55G replaces the MN102H55G mask ROM with the 128-kbyte EEPROM which is an electrically erasable/programmable memory. The MN102HF55G has two modes: PROM writer mode which uses a dedicated writer (either a DATA-I/O LabSite writer or a Minato Electronics Model 1930 writer) and onboard serial programming mode which the CPU controls.

The 128-kbyte flash memory is divided into three spaces as follows:

1. Load program area (1 kbyte: x'80000' - x'803FF')
   This area stores the load program for serial programming.

2. Fixed user program area (5 kbytes: x'80400' - x'817FF')
   This area stores the user program. It is programmed only in PROM writer mode.

3. User program area (122 kbytes: x'81800' - x'9FFFF')
   This area stores the user program. It is programmed in both PROM writer mode and onboard serial programming mode.

The operation is guaranteed with up to ten programming.

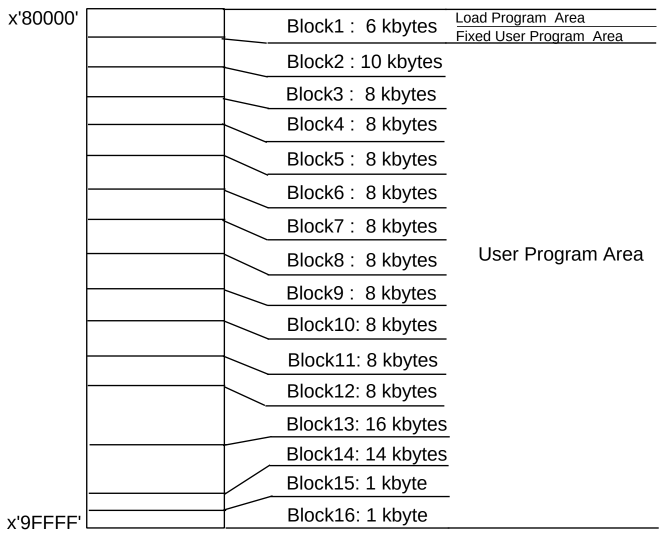


Figure 11-4-1 Memory Map for Flash EEPROM Version

11-4-2 Flash EEPROM Programming

The following figure shows the steps of flash memory programming.

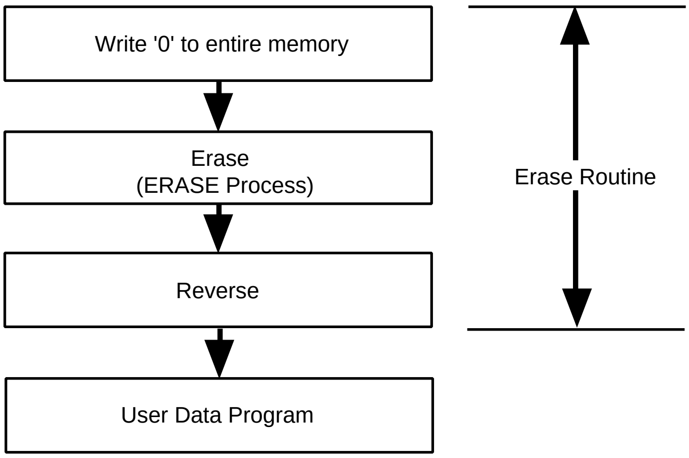


Figure 11-4-2 Flash EEPROM Program Flow

As the above figure shows, programming starts after erasing is completed. The whole erase routine consists of three steps:

1. Programming process which writes x'0000' to flash EEPROM before the actual erase process occurs
2. Erase process which operate the actual erasing
3. Reverse process.

11-4-3 PROM Writer Mode

In this mode, the MN102HF55G allows a PROM writer to program the flash EEPROM. The MN102HF55G uses a dedicated adaptor, which connects to the DATA-I/O's LabSite PROM writer or the Minato Electronics Model 1930 writer. (Using the dedicated adaptor selects PROM writer mode automatically.)

11-4-4 Onboard Serial Programming Mode

The serial programming mode is used to program the flash ROM in the MN102HF55G that is installed on the board. The following sections describe the MN102HF55G hardware, system configuration, protocol for this programming mode.

When using YDC dedicated writer, please refer to its user manual. The load program is attached to the serial writer.

11-4-5 Hardware Used in Serial Programming Mode

■ Interface

The MN102HF55G incorporates the following functions as I/F for serial programming.

- One 8-bit Serial Interface
  - ♦ Data transmission/reception synchronizing external clock or internal clock
  - ♦ Bit order: LSB first, MSB first
  - ♦ Maximum transfer speed: 7.5 Mbps (at 30-MHz oscillation)
  - ♦ Positive input/output logic

- Two Input/Output Pins
  - ♦ SBT, SBD reserved for serial interface

■ I/F Block Diagram

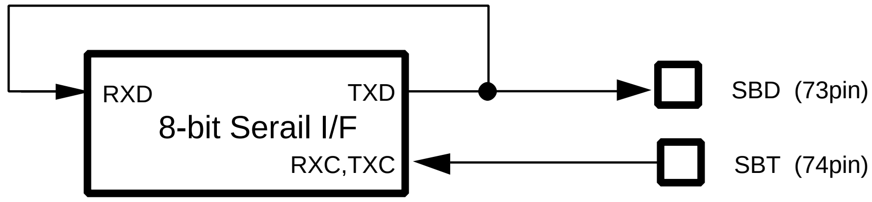


Figure 11-4-3 8-bit Serial Interface Block Diagram for Serial Writer

■ Memory Space of Internal Flash EEPROM

Address	Size	Area
x'80000' - x'803FF'	1 kbytes	Serial Writer Load Program Area
x'80400' - x'817FF'	5 kbytes	Fixed User Program Area
x'81800 - x'81807'	8 bytes	Security Code
x'81808' - x'8180F'	8 bytes	Reserved Area
x'81810' - x'81817'	8 Bytes	Branch Instruction to Reset Service Routine
x'81818' - x'8181F'	8 bytes	Branch Instruction to Interrupt Service Routine
x'81820' - x'9FFFF'	122 kbytes to 32 bytes	User Program Area

Figure 11-4-4 Flash EEPROM Memory Space

- Serial Writer Load Program Area
  - ♦ The 1-kbyte area from x'80000' stores the load program for serial writer.
  - ♦ In onboard serial programming mode, the erasing/programming in this area is protected. (Programming is possible by using the parallel writer.)

- Fixed User Program Area
  - ♦ The 5-kbyte area from x'80400' stores the fixed user program.
  - ♦ In onboard serial programming mode, the erasing/programming in this area is protected. (Programming is possible by using the parallel writer.)

- Security Code
  - ♦ The area stores the security code for the serial writer password.
  - ♦ Enter 8-character ASCII code.

- Reserved Area
  - ♦ Do not write in this area.

- Branch Instruction to Reset Start Service Routine
  - ♦ Normally, the reset start address is x'80000', but the program branches into x'81820' with the soft branch instruction in the serial writer loader. In this area, the JMP instruction to the actual reset service routine is stored.

- Branch Instruction to Interrupt Service Routine
  - ♦ Normally, the jump address at interrupt is x'80008', but the program branches into x'81818' with the soft branch instruction in the serial writer loader. In this area, the JMP instruction to the actual interrupt service routine is stored.

- User Program Area
  - ♦ This area stores the user program.
  - ♦ Size = 122kbytes - 32 bytes

11-4-6 Connecting Onboard Serial Programming Mode

Use YDC serial writer for flash microcontroller.
All input/output pins must be set to input at reset release.

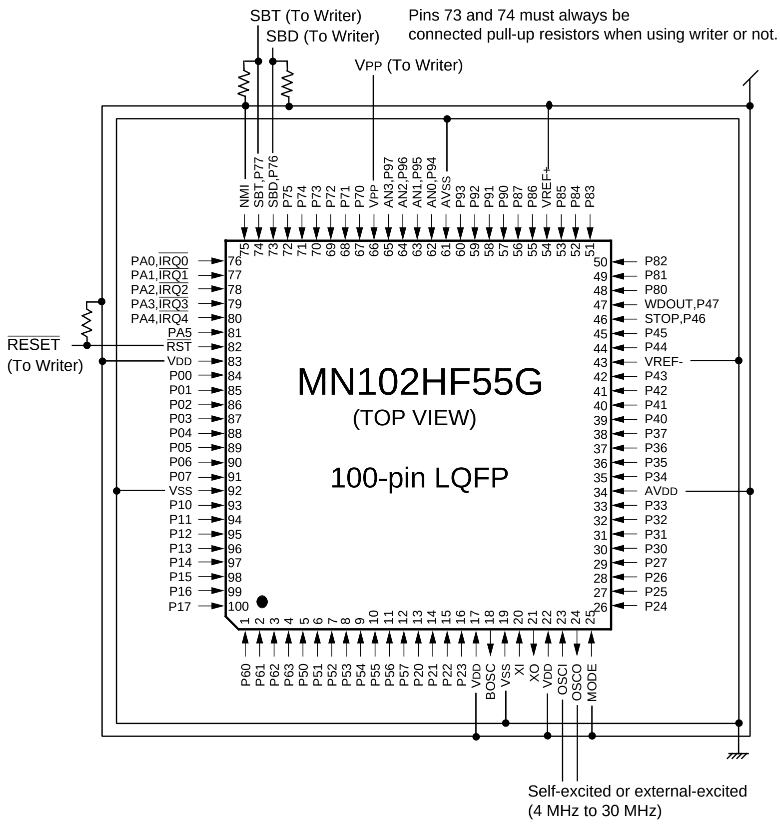


Figure 11-4-5 Pin Configuration During Serial Programming

Pins 73, 74 and 82 connect to the serial writer. VDD and Vss connect to the external power sources of 3.3 V and 0 V respectively. In addition, the level is detected by the writer, VDD and Vss must be output to the writer. OSCI and OSCO must be set to the self-excited oscillation or external excited oscillation. The input pins with no specifications in the above figure are 'don't care'. Fix them to VDD or Vss. The output pins (BOSC, XI) with no specifications in the above figure must be open.

11-4-7 System Configuration for Onboard Serial Programming

■ System Configuration

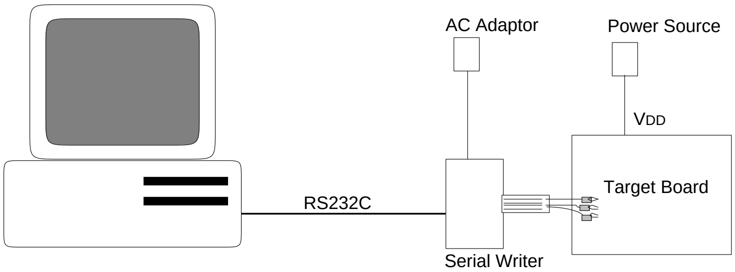


Figure 11-4-6 System Configuration for Onboard Serial Writer

The PC sends the program data to the serial writer through RS-232C. The serial writer programs the flash memory through serial communication between the serial writer and the MN102HF55G on the target board. The power is required only when the power source is supplied to the target.

■ Pin Connection for Target Board

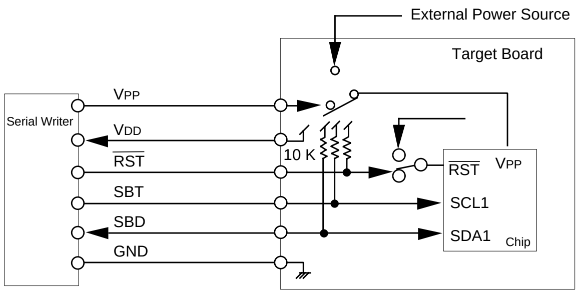


Figure 11-4-7 Target Board-Serial Writer Connection

■ Pin Description

♦ VPP : 5.0 V power supply (supplied from serial writer)
♦ VDD : 3.3 V external power supply
♦ VDD (for level detection) : VDD level detection pin for target board
♦ $\overline{\text{RST}}$: Reset
♦ SBT : Serial interface clock supply
♦ SBD : Serial interface data supply
♦ GND : Ground

• VDD detects the VDD level on the target board using the serial writer. If the VDD level is not satisfied, the serial writer outputs an error message.

• The serial writer supplies VPP. VPP for the serial writer and VPP for external power source for operation should be selected using switch.

• $\overline{\text{RST}}$ outputs microcontroller reset.

• Connect pullup resistors to $\overline{\text{RST}}$, SBT and SBD on the target board. The pullup resistor value is 10 kΩ ± 1 kΩ.

• $\overline{\text{RST}}$, SBT and SBD are output from the serial writer through an open collector.

■ MN102HF55G Clock on the Target Board

• Use the existing clock on the target board for the clock supply to the MN102HF55G on the target board. Because of this, the clock frequency of the MN102HF55G differs depending on each user purpose.

• The following table shows the clock frequency for the MN102HF55G during serial programming. The clock frequency for the MN102HF55G is assumed to be 30 MHz if the clock frequency is not specified in the manual. If the clock frequency for the MN102HF55G is different from the clock frequency on the target board, the value should be calculated proportionately depending on the clock frequency of the MN102HF55G.

Table 11-4-1 Clock Frequency

Max. Clock Frequency	Min. Clock Frequency
30 MHz	4 MHz

11-4-8 Onboard Serial Programming Mode Setup

■ Programming Mode Setup Timing

To set serial programming mode, the microcontroller must be in write mode.
This section describes the pin setup for the serial writer.

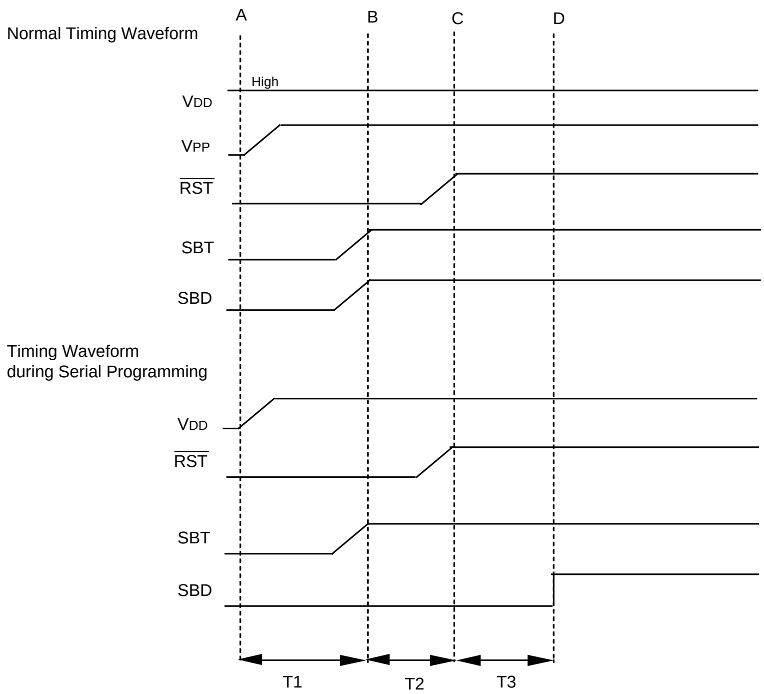


Figure 11-4-8 Timing for Onboard Serial Programming Mode

■ Setup Steps

1. Supply VDD from the external power. At this point, the serial writer detects the VDD level.
2. Supply VPP at Timing A. At this point, output $\overline{\text{RST}}$ = SBD = Low.
3. Through the serial writer, drive $\overline{\text{RST}}$ for T2 term from Timing B when SBT goes high while the MN102HF55G is on. The MN102HF55G initializes.
4. Through the serial writer, drive $\overline{\text{RST}}$ for T3 term from Timing C when SBD goes high while the MN102HF55G is on. This informs that the MN102HF55G is connected to the serial writer.
5. During T3 term, the serial writer makes SBD pin to input low level longer enough than the MN102HF55G stabilization wait time.

■ Load Program

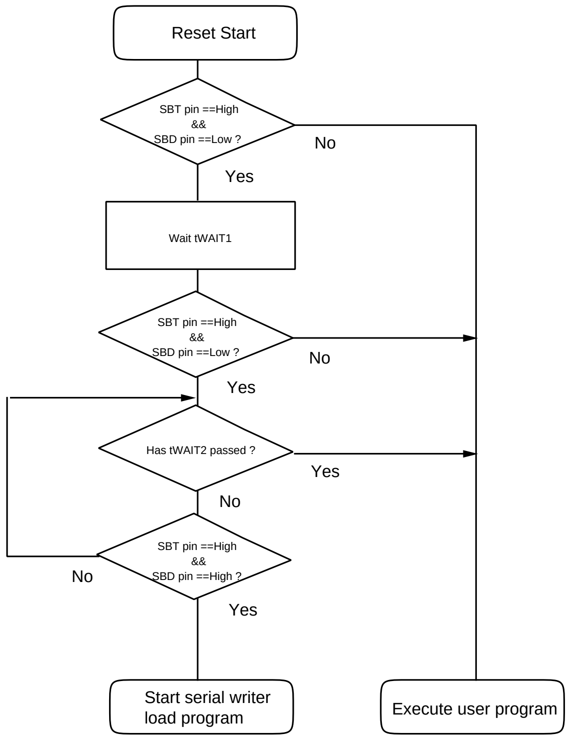


Figure 11-4-9 Load Program Start Flow

Conditions

1. When the load program initializes a reset start, SBD = low and SBT = high.
2. The program waits for tWAIT1.
3. SBD must still be low and SBT high.
4. Wait that both SBD and SBT become high during tWAIT2.

If any above conditions are not met, the program returns to the user program.

11-4-9 Branch to User Program

■ Branch to Reset Service Routine

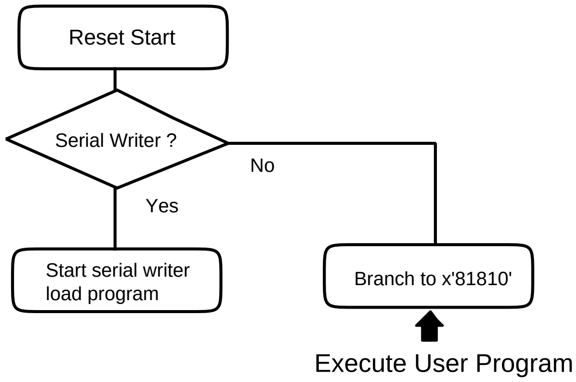


Figure 11-4-10 Reset Service Routine Flow

When the reset starts, the serial writer load program initializes only if SBD is low. The program branches to the user program at address x'81810'.

■ Branch to Interrupt Service Routine

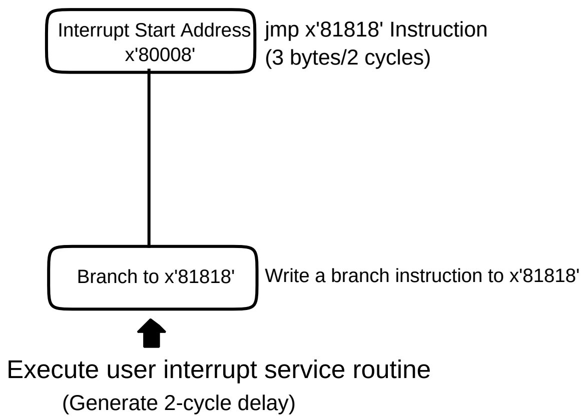


Figure 11-4-11 Interrupt Service Routine Flow

Write only the instruction branching to address x'81818' at the interrupt start address (x'80008').

11-4-10 Serial Interface for Onboard Serial Programming

■ Features

Fixed-length Serial Interface

• Character length	8 bits or 7 bits
• Transmission bit order	LSB/MSB (can be selected only when the character length is 8 bits.)
• Clock source	External clock, Timer 5 underflow (1/2, 1/8), Timer 1 underflow (1/8)
• Maximum transfer speed	7.5 Mbps (with a 30-MHz oscillator)
• Error detection	Overrun error
• Buffer	Transmit/receive shared buffer Single transmit buffer, Double receive buffer

■ Data Timing

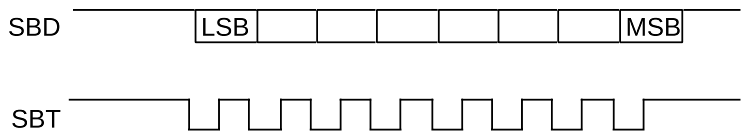


Figure 11-4-12 Data Transfer Timing

11-4-11 PROM Writer/Onboard Serial Programming

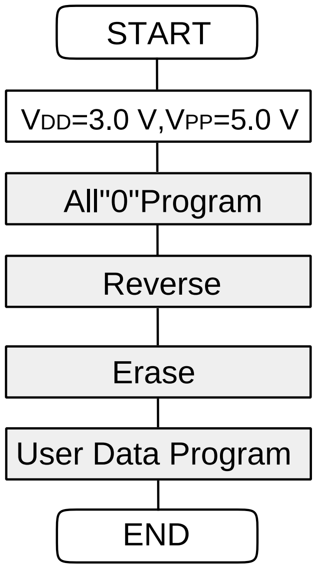


Figure 11-4-13 Programming Flow

11-5 List of MN102H00 Series Linear Address High-speed Edition Instructions

MN102H00 SERIES INSTRUCTION SET

Instruction	Mnemonic	Operation	OP	Flag								Code	Cycle	❋ Cycle	Machine Code
			EX.	VX	CX	NX	ZX	VF	CF	NF	ZF	Size			
MOV	MOV Dm,An	Dm→An	—	—	—	—	—	—	—	—	—	2	2	1	F2:30+Dm<<2+An
	MOV An,Dm	An→Dm	—	—	—	—	—	—	—	—	—	2	2	1	F2:F0+An<<2+Dm
	MOV Dn,Dm	Dn→Dm	—	—	—	—	—	—	—	—	—	1	1		80+Dn<<2+Dm *1
	MOV An,Am	An→Am	—	—	—	—	—	—	—	—	—	2	2	1	F2:70+An<<2+Am
	MOV PSW,Dn	PSW→Dn	0	—	—	—	—	—	—	—	—	2	2	1	F3:F0+Dn
	MOV Dn,PSW	Dn→PSW	—	●	●	●	●	●	●	●	●	2	3	2	F3:D0+Dn<<2
	MOV MDR,Dn	MDR→Dn	0	—	—	—	—	—	—	—	—	2	2	1	F3:E0+Dn
	MOV Dn,MDR	Dn→MDR	—	—	—	—	—	—	—	—	—	2	2	1	F3:C0+Dn<<2
	MOV (An),Dm	mem16(An)→Dm	S	—	—	—	—	—	—	—	—	1	1		20+An<<2+Dm
	MOV (d8,An),Dm	mem16(An+d8)→Dm	S	—	—	—	—	—	—	—	—	2	1		60+An<<2+Dm:d8
	MOV (d16,An),Dm	mem16(An+d16)→Dm	S	—	—	—	—	—	—	—	—	4	2	1	F7:C0+An<<2+Dm:d16-l:d16-h
	MOV (d24,An),Dm	mem16(An+d24)→Dm	S	—	—	—	—	—	—	—	—	5	3	2	F4:80+An<<2+Dm:d24-l:d24-m:d24-h
	MOV (Di,An),Dm	mem16(An+Di)→Dm	S	—	—	—	—	—	—	—	—	2	2	1	F1:40+Di<<4+An<<2+Dm
	MOV (abs16),Dn	mem16(abs16)→Dn	S	—	—	—	—	—	—	—	—	3	1		C8+Dn:abs16-l:abs16-h
	MOV (abs24),Dn	mem16(abs24)→Dn	S	—	—	—	—	—	—	—	—	5	3	2	F4:C0+Dn:abs24-l:abs24-m:abs24-h
	MOV (An),Am	mem24(An)→Am	—	—	—	—	—	—	—	—	—	2	2		70+An<<2+Am:00 *2
	MOV (d8,An),Am	mem24(An+d8)→Am	—	—	—	—	—	—	—	—	—	2	2		70+An<<2+Am:d8
	MOV (d16,An),Am	mem24(An+d16)→Am	—	—	—	—	—	—	—	—	—	4	3	2	F7:B0+An<<2+Am:d16-l:d16-h
	MOV (d24,An),Am	mem24(An+d24)→Am	—	—	—	—	—	—	—	—	—	5	4	3	F4:F0+An<<2+Am:d24-l:d24-m:d24-h
	MOV (Di,An),Am	mem24(An+Di)→Am	—	—	—	—	—	—	—	—	—	2	3	2	F1:00+Di<<4+An<<2+Am
	MOV (abs16),An	mem24(abs16)→An	—	—	—	—	—	—	—	—	—	4	3	2	F7:30+An:abs16-l:abs16-h
	MOV (abs24),An	mem24(abs24)→An	—	—	—	—	—	—	—	—	—	5	4	3	F4:D0+An:abs24-l:abs24-m:abs24-h
	MOV Dm,(An)	Dm→mem16(An)	—	—	—	—	—	—	—	—	—	1	1		00+An<<2+Dm
	MOV Dm,(d8,An)	Dm→mem16(An+d8)	—	—	—	—	—	—	—	—	—	2	1		40+An<<2+Dm:d8
	MOV Dm,(d16,An)	Dm→mem16(An+d16)	—	—	—	—	—	—	—	—	—	4	2	1	F7:80+An<<2+Dm:d16-l:d16-h
	MOV Dm,(d24,An)	Dm→mem16(An+d24)	—	—	—	—	—	—	—	—	—	5	3	2	F4:00+An<<2+Dm:d24-l:d24-m:d24-h
	MOV Dm,(Di,An)	Dm→mem16(An+Di)	—	—	—	—	—	—	—	—	—	2	2	1	F1:C0+Di<<4+An<<2+Dm
	MOV Dn,(abs16)	Dn→mem16(abs16)	—	—	—	—	—	—	—	—	—	3	1		C0+Dn:abs16-l:abs16-h
	MOV Dn,(abs24)	Dn→mem16(abs24)	—	—	—	—	—	—	—	—	—	5	3	2	F4:40+Dn:abs24-l:abs24-m:abs24-h
	MOV Am,(An)	Am→mem24(An)	—	—	—	—	—	—	—	—	—	2	2		50+An<<2+Am:00 *3
	MOV Am,(d8,An)	Am→mem24(An+d8)	—	—	—	—	—	—	—	—	—	2	2		50+An<<2+Am:d8
	MOV Am,(d16,An)	Am→mem24(An+d16)	—	—	—	—	—	—	—	—	—	4	3	2	F7:A0+An<<2+Am:d16-l:d16-h
	MOV Am,(d24,An)	Am→mem24(An+d24)	—	—	—	—	—	—	—	—	—	5	4	3	F4:10+An<<2+Am:d24-l:d24-m:d24-h
	MOV Am,(Di,An)	Am→mem24(An+Di)	—	—	—	—	—	—	—	—	—	2	3	2	F1:80+Di<<4+An<<2+Am
	MOV An,(abs16)	An→mem24(abs16)	—	—	—	—	—	—	—	—	—	4	3	2	F7:20+An:abs16-l:abs16-h
	MOV An,(abs24)	An→mem24(abs24)	—	—	—	—	—	—	—	—	—	5	4	3	F4:50+An:abs24-l:abs24-m:abs24-h
	MOV imm8,Dn	imm8→Dn	S	—	—	—	—	—	—	—	—	2	1		80+Dn<<2+Dn:imm8
	MOV imm16,Dn	imm16→Dn	S	—	—	—	—	—	—	—	—	3	1		F8+Dn:imm16-l:imm16-h
	MOV imm24,Dn	imm24→Dn	—	—	—	—	—	—	—	—	—	5	3	2	F4:70+Dn:imm24-l:imm24-m:imm24-h
	MOV imm16,An	imm16→An	0	—	—	—	—	—	—	—	—	3	1		DC+An:imm16-l:imm16-h
	MOV imm24,An	imm24→An	—	—	—	—	—	—	—	—	—	5	3	2	F4:74+An:imm24-l:imm24-m:imm24-h
MOVX	MOVX (d8,An),Dm	mem24(An+d8)→Dm	—	—	—	—	—	—	—	—	—	3	3	2	F5:70+An<<2+Dm:d8
	MOVX (d16,An),Dm	mem24(An+d16)→Dm	—	—	—	—	—	—	—	—	—	4	3	2	F7:70+An<<2+Dm:d16-l:d16-h
	MOVX (d24,An),Dm	mem24(An+d24)→Dm	—	—	—	—	—	—	—	—	—	5	4	3	F4:B0+An<<2+Dm:d24-l:d24-m:d24-h
	MOVX Dm,(d8,An)	Dm→mem24(An+d8)	—	—	—	—	—	—	—	—	—	3	3	2	F5:50+An<<2+Dm:d8
	MOVX Dm,(d16,An)	Dm→mem24(An+d16)	—	—	—	—	—	—	—	—	—	4	3	2	F7:60+An<<2+Dm:d16-l:d16-h
	MOVX Dm,(d24,An)	Dm→mem24(An+d24)	—	—	—	—	—	—	—	—	—	5	4	3	F4:30+An<<2+Dm:d24-l:d24-m:d24-h
MOVB	MOVB (An),Dm	mem8(An)→Dm	S	—	—	—	—	—	—	—	—	2	2		30+An<<2+Dm:B8+Dn *4
	MOVB (d8,An),Dm	mem8(An+d8)→Dm	S	—	—	—	—	—	—	—	—	3	2	1	F5:20+An<<2+Dm:d8
	MOVB (d16,An),Dm	mem8(An+d16)→Dm	S	—	—	—	—	—	—	—	—	4	2	1	F7:D0+An<<2+Dm:d16-l:d16-h
	MOVB (d24,An),Dm	mem8(An+d24)→Dm	S	—	—	—	—	—	—	—	—	5	3	2	F4:A0+An<<2+Dm:d24-l:d24-m:d24-h
	MOVB (Di,An),Dm	mem8(An+Di)→Dm	S	—	—	—	—	—	—	—	—	2	2	1	F0:40+Di<<4+An<<2+Dm
	MOVB (abs16),Dn	mem8(abs16)→Dn	S	—	—	—	—	—	—	—	—	4	2		CC+Dn:abs16-l:abs16-h:B8+Dn *5
	MOVB (abs24),Dn	mem8(abs24)→Dn	S	—	—	—	—	—	—	—	—	5	3	2	F4:C4+Dn:abs24-l:abs24-m:abs24-h
	MOVB Dm,(An)	Dm→mem8(An)	—	—	—	—	—	—	—	—	—	1	1		10+Dm<<2+An
	MOVB Dm,(d8,An)	Dm→mem8(An+d8)	—	—	—	—	—	—	—	—	—	3	2	1	F5:10+An<<2+Dm:d8
	MOVB Dm,(d16,An)	Dm→mem8(An+d16)	—	—	—	—	—	—	—	—	—	4	2	1	F7:90+An<<2+Dm:d16-l:d16-h
	MOVB Dm,(d24,An)	Dm→mem8(An+d24)	—	—	—	—	—	—	—	—	—	5	3	2	F4:20+An<<2+Dm:d24-l:d24-m:d24-h
	MOVB Dm,(Di,An)	Dm→mem8(An+Di)	—	—	—	—	—	—	—	—	—	2	2	1	F0:C0+Di<<4+An<<2+Dm

❋ Quick decoder : ON
(This setting cannot be made in this series.)

Notes:
1* It is not possible to specify that Dn=Dm.
2* This instruction is supported by the assembler. For "MOV (d8,An),Am" the assembler will generate a bit pattern for d8=0.
3* This instruction is supported by the assembler. For "MOV Am,(d8,An)" the assembler will generate a bit pattern for d8=0.
4* This instruction is supported by the assembler. The assembler generates bit patterns for the two instructions "MOVBU (An),Dm" and "EXTXB Dm".
5* This instruction is supported by the assembler. The assembler generates bit patterns for the two instructions "MOVBU (abs16),Dn" and "EXTXB Dn".

Instruction	Mnemonic	Operation	OP EX.	Flag								Code Size	Cycle	❋ Cycle	Machine Code
				VX	CX	NX	ZX	VF	CF	NF	ZF				
MOVB	MOVB Dn,(abs16)	Dn→mem8(abs16)	—	—	—	—	—	—	—	—	—	3	1		C4+Dn:abs16-l:abs16-h
	MOVB Dn,(abs24)	Dn→mem8(abs24)	—	—	—	—	—	—	—	—	—	5	3	2	F4:44+Dn:abs24-l:abs24-m:abs24-h
MOVBU	MOVBU (An),Dm	mem8(An)→Dm	0	—	—	—	—	—	—	—	—	1	1		30+An<<2+Dm
	MOVBU (d8,An),Dm	mem8(An+d8)→Dm	0	—	—	—	—	—	—	—	—	3	2	1	F5:30+An<<2+Dm:d8
	MOVBU (d16,An),Dm	mem8(An+d16)→Dm	0	—	—	—	—	—	—	—	—	4	2	1	F7:50+An<<2+Dm:d16-l:d16-h
	MOVBU (d24,An),Dm	mem8(An+d24)→Dm	0	—	—	—	—	—	—	—	—	5	3	2	F4:90+An<<2+Dm:d24-l:d24-m:d24-h
	MOVBU (Di,An),Dm	mem8(An+Di)→Dm	0	—	—	—	—	—	—	—	—	2	2	1	F0:80+Di<<4+An<<2+Dm
	MOVBU (abs16),Dn	mem8(abs16)→Dn	0	—	—	—	—	—	—	—	—	3	1		CC+Dn:abs16-l:abs16-h
	MOVBU (abs24),Dn	mem8(abs24)→Dn	0	—	—	—	—	—	—	—	—	5	3	2	F4:C8+Dn:abs24-l:abs24-m:abs24-h
EXT	EXT Dn	If Dn.bp15=0, x'0000'→MDR If Dn.bp15=1, x'FFFF'→MDR	S	—	—	—	—	—	—	—	—	2	3	2	F3:C1+Dn<<2 *6
EXTX	EXTX Dn	If Dn.bp15=0, Dn&x'00FFFF'→Dn If Dn.bp15=1, Dn l x'FF0000'→Dn	S	—	—	—	—	—	—	—	—	1	1		B0+Dn *7
EXTXU	EXTXU Dn	Dn&x'00FFFF'→Dn	0	—	—	—	—	—	—	—	—	1	1		B4+Dn *8
EXTXB	EXTXB Dn	If Dn.bp7=0, Dn&x'0000FF'→Dn If Dn.bp7=1, Dn l x'FFFF00'→Dn	S	—	—	—	—	—	—	—	—	1	1		B8+Dn *9
EXTXBU	EXTXBU Dn	Dn&x'0000FF'→Dn	0	—	—	—	—	—	—	—	—	1	1		BC+Dn *10
ADD	ADD Dn,Dm	Dm+Dn→Dm	—	●	●	●	●	●	●	●	●	1	1		90+Dn<<2+Dm
	ADD Dm,An	An+Dm→An	—	●	●	●	●	●	●	●	●	2	2	1	F2:00+Dm<<2+An
	ADD An,Dm	Dm+An→Dm	—	●	●	●	●	●	●	●	●	2	2	1	F2:C0+An<<2+Dm
	ADD An,Am	Am+An→Am	—	●	●	●	●	●	●	●	●	2	2	1	F2:40+An<<2+Am
	ADD imm8,Dn	Dn+imm8→Dn	S	●	●	●	●	●	●	●	●	2	1		D4+Dn:imm8
	ADD imm16,Dn	Dn+imm16→Dn	S	●	●	●	●	●	●	●	●	4	2	1	F7:18+Dn:imm16-l:imm16-h
	ADD imm24,Dn	Dn+imm24→Dn	—	●	●	●	●	●	●	●	●	5	3	2	F4:60+Dn:imm24-l:imm24-m:imm24-h
	ADD imm8,An	An+imm8→An	S	●	●	●	●	●	●	●	●	2	1		D0+An:imm8
	ADD imm16,An	An+imm16→An	S	●	●	●	●	●	●	●	●	4	2	1	F7:08+An:imm16-l:imm16-h
	ADD imm24,An	An+imm24→An	—	●	●	●	●	●	●	●	●	5	3	2	F4:64+An:imm24-l:imm24-m:imm24-h
ADDC	ADDC Dn,Dm	Dm+Dn+CF→Dm	—	●	●	●	●	●	●	●	●	2	2	1	F2:80+Dn<<2+Dm
ADDNF	ADDNF imm8,An	An+imm8→An	S	—	—	—	—	—	—	—	—	3	2	1	F5:0C+An:imm8 *11
SUB	SUB Dn,Dm	Dm-Dn→Dm	—	●	●	●	●	●	●	●	●	1	1		A0+Dn<<2+Dm
	SUB Dm,An	An-Dm→An	—	●	●	●	●	●	●	●	●	2	2	1	F2:10+Dm<<2+An
	SUB An,Dm	Dm-An→Dm	—	●	●	●	●	●	●	●	●	2	2	1	F2:D0+An<<2+Dm
	SUB An,Am	Am-An→Am	—	●	●	●	●	●	●	●	●	2	2	1	F2:50+An<<2+Am
	SUB imm16,Dn	Dn-imm16→Dn	S	●	●	●	●	●	●	●	●	4	2	1	F7:1C+Dn:imm16-l:imm16-h
	SUB imm24,Dn	Dn-imm24→Dn	—	●	●	●	●	●	●	●	●	5	3	2	F4:68+Dn:imm24-l:imm24-m:imm24-h
	SUB imm16,An	An-imm16→An	S	●	●	●	●	●	●	●	●	4	2	1	F7:0C+An:imm16-l:imm16-h
	SUB imm24,An	An-imm24→An	—	●	●	●	●	●	●	●	●	5	3	2	F4:6C+An:imm24-l:imm24-m:imm24-h
SUBC	SUBC Dn,Dm	Dm-Dn-CF→Dm	—	●	●	●	●	●	●	●	●	2	2	1	F2:90+Dn<<2+Dm
MUL	MUL Dn,Dm	Dm ✻ Dn→Dm (Dm ✻ Dn)>>16→MDR	—	?	?	?	?	0	?	●	●	2	12	11	F3:40+Dn<<2+Dm *12
MULU	MULU Dn,Dm	Dm ✻ Dn→Dm (Dm ✻ Dn)>>16→MDR	—	?	?	?	?	0	?	●	●	2	12	11	F3:50+Dn<<2+Dm *13
MULQ	MULQ Dn,Dm	Dm ✻ Dn→Dm (H.M.) (Dm ✻ Dn)>>16→MDR	—	—	—	—	—	—	—	—	—	3	3	2	F5:60+Dn<<2+Dm:10
MULQL	MULQL Dn,Dm	Dm ✻ Dn→Dm (H.M.)	—	—	—	—	—	—	—	—	—	3	2	1	F5:40+Dn<<2+Dm:00
	MULQL imm8,Dn	Dn ✻ imm8→Dn (H.M.)	—	—	—	—	—	—	—	—	—	4	2	1	F5:F0+Dn:04:imm8
	MULQL imm16,Dn	Dn ✻ imm16→Dn (H.M.)	—	—	—	—	—	—	—	—	—	5	3	2	F5:F4+Dn:08:imm16-l:imm16-h
MULQH	MULQH Dn,Dm	(Dm ✻ Dn)>>16→Dm (H.M.)	S	—	—	—	—	—	—	—	—	3	2	1	F5:40+Dn<<2+Dm:01
	MULQH imm8,Dn	(Dn ✻ imm8)>>16→Dn (H.M.)	S	—	—	—	—	—	—	—	—	4	2	1	F5:F0+Dn:05:imm8
	MULQH imm16,Dn	(Dn ✻ imm16)>>16→Dn (H.M.)	S	—	—	—	—	—	—	—	—	5	3	2	F5:F4+Dn:09:imm16-l:imm16-h
DIVU	DIVU Dn,Dm	(MDR<<16+Dm)/Dn→Dm ...MDR	—	?	?	0/?	●/?	0/1	?	●/?	●/?	2	13	12	F3:60+Dn<<2+Dm *14

Notes:
6* 32-bit sign extended word data
7* 24-bit sign extended word data
8* 24-bit zero extended word data
9* 24-bit sign extended byte data
10* 24-bit zero extended byte data
11* Addition without changing flag
12* 16x16 = 32 (signed)
13* 16x16 = 32 (unsigned)
14* 32÷16 = 16…16 (unsigned)

❋ Quick decoder : ON
(This setting cannot be made in this series.)

Instruction	Mnemonic	Operation	OP	Flag								Code	Cycle	❋	Machine Code
			EX.	VX	CX	NX	ZX	VF	CF	NF	ZF	Size		Cycle	
CMP	CMP Dn,Dm	Dm-Dn...PSW	—	●	●	●	●	●	●	●	●	2	2	1	F3:90+Dn<<2+Dm
	CMP Dm,An	An-Dm...PSW	—	●	●	●	●	●	●	●	●	2	2	1	F2:20+Dm<<2+An
	CMP An,Dm	Dm-An...PSW	—	●	●	●	●	●	●	●	●	2	2	1	F2:E0+An<<2+Dm
	CMP An,Am	Am-An...PSW	—	●	●	●	●	●	●	●	●	2	2	1	F2:60+An<<2+Am
	CMP imm8,Dn	Dn-imm8...PSW	S	●	●	●	●	●	●	●	●	2	1		D8+Dn:imm8
	CMP imm16,Dn	Dn-imm16...PSW	S	●	●	●	●	●	●	●	●	4	2	1	F7:48+Dn:imm16-l:imm16-h
	CMP imm24,Dn	Dn-imm24...PSW	—	●	●	●	●	●	●	●	●	5	3	2	F4:78+Dn:imm24-l:imm24-m:imm24-h
	CMP imm16,An	An-imm16...PSW	0	●	●	●	●	●	●	●	●	3	1		EC+An:imm16-l:imm16-h
	CMP imm24,An	An-imm24...PSW	—	●	●	●	●	●	●	●	●	5	3	2	F4:7C+An:imm24-l:imm24-m:imm24-h
AND	AND Dn,Dm	Dm&(x'FF0000' l Dn)→Dm	—	—	—	—	—	0	0	●	●	2	2	1	F3:00+Dn<<2+Dm *15
	AND imm8,Dn	Dn&(x'FF0000' l imm8)→Dn	0	—	—	—	—	0	0	●	●	3	2	1	F5:00+Dn:imm8 *15
	AND imm16,Dn	Dn&(x'FF0000' l imm16)→Dn	—	—	—	—	—	0	0	●	●	4	2	1	F7:00+Dn:imm16-l:imm16-h *15
	AND imm16,PSW	PSW&imm16→PSW	—	●	●	●	●	●	●	●	●	4	3	2	F7:10:imm16-l:imm16-h *15
OR	OR Dn,Dm	Dm l (Dn&x'00FFFF')→Dm	—	—	—	—	—	0	0	●	●	2	2	1	F3:10+Dn<<2+Dm *15
	OR imm8,Dn	Dn l imm8→Dn	0	—	—	—	—	0	0	●	●	3	2	1	F5:08+Dn:imm8 *15
	OR imm16,Dn	Dn l imm16→Dn	—	—	—	—	—	0	0	●	●	4	2	1	F7:40+Dn:imm16-l:imm16-h *15
	OR imm16,PSW	PSW l imm16→PSW	—	●	●	●	●	●	●	●	●	4	3	2	F7:14:imm16-l:imm16-h *15
XOR	XOR Dn,Dm	Dm^(x'00FFFF'&Dn)→Dm	—	—	—	—	—	0	0	●	●	2	2	1	F3:20+Dn<<2+Dm *15
	XOR imm16,Dn	Dn^imm16→Dn	—	—	—	—	—	0	0	●	●	4	2	1	F7:4C+Dn:imm16-l:imm16-h *15
NOT	NOT Dn	Dn^x'00FFFF'→Dn	—	—	—	—	—	0	0	●	●	2	2	1	F3:E4+Dn *15
ASR	ASR Dn	Dn.lsb→CF Dn.bp→Dn.bp-1(bp15~1) Dn.bp15→Dn.bp15	—	—	—	—	—	0	●	●	●	2	2	1	F3:38+Dn *15
LSR	LSR Dn	Dn.lsb→CF Dn.bp→Dn.bp-1(bp15~1) 0→Dn.bp15	—	—	—	—	—	0	●	0	●	2	2	1	F3:3C+Dn *15
ROR	ROR Dn	Dn.lsb→temp Dn.bp→Dn.bp-1(bp15~1) CF→Dn.bp15 temp→CF	—	—	—	—	—	0	●	●	●	2	2	1	F3:34+Dn *15
ROL	ROL Dn	Dn.bp15→temp Dn.bp→Dn.bp+1(bp14~0) CF→Dn.lsb temp→CF	—	—	—	—	—	0	●	●	●	2	2	1	F3:30+Dn *15
BTST	BTST imm8,Dn	Dn&imm8...PSW	0	—	—	—	—	0	0	0	●	3	2	1	F5:04+Dn:imm8
	BTST imm16,Dn	Dn&imm16...PSW	0	—	—	—	—	0	0	●	●	4	2	1	F7:04+Dn:imm16-l:imm16-h
BSET	BSET Dm,(An)	mem8(An)&Dm...PSW mem8(An) l Dm→mem8(An)	0	—	—	—	—	0	0	0	●	2	5	4	F0:20+An<<2+Dm *16
	BSET imm8,(abs16)	mem8(abs16) l imm8 →mem8(abs16)	—	—	—	—	—	—	—	—	—	5	4	3	F4:E3:abs16-l:abs16-h:imm8
	BSET imm8,(abs24)	mem8(abs24) l imm8 →mem8(abs24)	—	—	—	—	—	—	—	—	—	6	5	4	F4:4B:abs24-l:abs24-m:abs24-h:imm8
	BSET imm8,(d8,An)	mem8(An+d8) l imm8 →mem8(An+d8)	—	—	—	—	—	—	—	—	—	4	4	3	F4:E8+An:d8:imm8
	BSET (abs16)bp	mem8(abs16) l (1<<bp) →mem8(abs16)	—	—	—	—	—	—	—	—	—	4	4	3	F5:D0+bp:abs16-l:abs16-h
	BSET (abs24)bp	mem8(abs24) l (1<<bp) →mem8(abs24)	—	—	—	—	—	—	—	—	—	6	6	5	F3:FE:D0+bp:abs24-l:abs24-m:abs24-h
	BSET (d8,An)bp	mem8(An+d8) l (1<<bp) →mem8(An+d8)	—	—	—	—	—	—	—	—	—	3 3 4 4	4 4 5 5	3 3 4 4	F5:90+bp:d8 An=A0 F5:98+bp:d8 An=A1 F3:FF:90+bp:d8 An=A2 F3:FF:98+bp:d8 An=A3

❋ Quick decoder : ON
(This setting cannot be made in this series.)

Notes: 15* 16-bit computation
16* Performed under the conditions of bus lock and disabled interrupts.

Instruction	Mnemonic	Operation	OP EX.	Flag VX	CX	NX	ZX	VF	CF	NF	ZF	Code Size	Cycle	❋ Cycle	Machine Code
BCLR	BCLR Dm,(An)	mem8(An)&Dm...PSW mem8(An)&(¯Dm)→mem8(An)	0	—	—	—	—	0	0	0	●	2	5	4	F0:30+An<<2+Dm *16
	BCLR imm8,(abs16)	mem8(abs16)&(¯imm8) →mem8(abs16)	—	—	—	—	—	—	—	—	—	5	4	3	F4:E7:abs16-l:abs16-h:imm8
	BCLR imm8,(abs24)	mem8(abs24)&(¯imm8) →mem8(abs24)	—	—	—	—	—	—	—	—	—	6	5	4	F4:4F:abs24-l:abs24-m:abs24-h:imm8
	BCLR imm8,(d8,An)	mem8(An+d8)&(¯imm8) →mem8(An+d8)	—	—	—	—	—	—	—	—	—	4	4	3	F4:EC+An:d8:imm8
	BCLR (abs16)bp	mem8(abs16)&¯(1<<bp) →mem8(abs16)	—	—	—	—	—	—	—	—	—	4	4	3	F5:D8+bp:abs16-l:abs16-h
	BCLR (abs24)bp	mem8(abs24)&¯(1<<bp) →mem8(abs24)	—	—	—	—	—	—	—	—	—	6	6	5	F3:FE:D8+bp:abs24-l:abs24-m:abs24-h
	BCLR (d8,An)bp	mem8(An+d8)&¯(1<<bp) →mem8(An+d8)	—	—	—	—	—	—	—	—	—	3 3 4 4	4 4 5 5	3 3 4 4	F5:B0+bp:d8 An=A0 F5:B8+bp:d8 An=A1 F3:FF:B0+bp:d8 An=A2 F3:FF:B8+bp:d8 An=A3
TBZ	TBZ (abs16)bp,label	mem8(abs16)&(1<<bp)...PSW If ZF=1, PC+5+d8(label)→PC If ZF=0, PC+5→PC	—	—	—	—	—	0	0	0	●	5	5/4	4/3	F5:C0+bp:abs16-l:abs16-h:label
	TBZ (abs24)bp,label	mem8(abs24)&(1<<bp)...PSW If ZF=1, PC+7+d8(label)→PC If ZF=0, PC+7→PC	—	—	—	—	—	0	0	0	●	7	7/6	6/5	F3:FE:C0+bp:abs24-l:abs24-m: abs24-h:label
	TBZ (d8,An)bp,label	mem8(An+d8)&(1<<bp)...PSW If ZF=1, PC+4(5)+d8(label)→PC If ZF=0, PC+4(5)→PC	—	—	—	—	—	0	0	0	●	4 4 5 5	5/4 5/4 6/5 6/5	4/3 4/3 5/4 5/4	F5:80+bp:d8:label An=A0 F5:88+bp:d8:label An=A1 F3:FF:80+bp:d8:label An=A2 F3:FF:88+bp:d8:label An=A3
TBNZ	TBNZ (abs16)bp,label	mem8(abs16)&(1<<bp)...PSW If ZF=1, PC+5→PC If ZF=0, PC+5+d8(label)→PC	—	—	—	—	—	0	0	0	●	5	5/4	4/3	F5:C8+bp:abs16-l:abs16-h:label
	TBNZ (abs24)bp,label	mem8(abs24)&(1<<bp)...PSW If ZF=1, PC+7→PC If ZF=0, PC+7+d8(label)→PC	—	—	—	—	—	0	0	0	●	7	7/6	6/5	F3:FE:C8+bp:abs24-l:abs24-m: abs24-h:label
	TBNZ (d8,An)bp,label	mem8(An+d8)&(1<<bp)...PSW If ZF=1, PC+4(5)→PC If ZF=0, PC+4(5)+d8(label)→PC	—	—	—	—	—	0	0	0	●	4 4 5 5	5/4 5/4 6/5 6/5	4/3 4/3 5/4 5/4	F5:A0+bp:d8:label An=A0 F5:A8+bp:d8:label An=A1 F3:FF:A0+bp:d8:label An=A2 F3:FF:A8+bp:d8:label An=A3
Bcc	BEQ label	If ZF=1, PC+2+d8(label)→PC If ZF=0, PC+2→PC	—	—	—	—	—	—	—	—	—	2	2/1		E8:d8 *17
	BNE label	If ZF=0, PC+2+d8(label)→PC If ZF=1, PC+2→PC	—	—	—	—	—	—	—	—	—	2	2/1		E9:d8 *18
	BLT label	If (VF^NF)=1, PC+2+d8(label)→PC If (VF^NF)=0, PC+2→PC	—	—	—	—	—	—	—	—	—	2	2/1		E0:d8 *19

❋ Quick decoder : ON
(This setting cannot be made in this series.)

Notes: 16* Performed under the conditions of bus lock and disabled interrupts.
17* src=dest (lower 16 bits)
18* src≠dest (lower 16 bits)
19* src>dest (lower 16 bits, signed)

Instruction	Mnemonic	Operation	OP EX.	Flag								Code Size	Cycle	※ Cycle	Machine Code
				VX	CX	NX	ZX	VF	CF	NF	ZF				
Bcc	BLE label	If ((VF^NF) l ZF)=1, PC+2+d8(label)→PC If ((VF^NF) l ZF)=0, PC+2→PC	—	—	—	—	—	—	—	—	—	2	2/1		E3:d8 *20
	BGE label	If (VF^NF)=0, PC+2+d8(label)→PC If (VF^NF)=1, PC+2→PC	—	—	—	—	—	—	—	—	—	2	2/1		E2:d8 *21
	BGT label	If ((VF^NF) l ZF)=0, PC+2+d8(label)→PC If ((VF^NF) l ZF)=1, PC+2→P	—	—	—	—	—	—	—	—	—	2	2/1		E1:d8 *22
	BCS label	If CF=1, PC+2+d8(label)→PC If CF=0, PC+2→PC	—	—	—	—	—	—	—	—	—	2	2/1		E4:d8 *23
	BLS label	If (CF l ZF)=1, PC+2+d8(label)→PC If (CF l ZF)=0, PC+2→PC	—	—	—	—	—	—	—	—	—	2	2/1		E7:d8 *24
	BCC label	If CF=0, PC+2+d8(label)→PC If CF=1, PC+2→PC	—	—	—	—	—	—	—	—	—	2	2/1		E6:d8 *25
	BHI label	If (CF l ZF)=0, PC+2+d8(label)→PC If (CF l ZF)=1, PC+2→PC	—	—	—	—	—	—	—	—	—	2	2/1		E5:d8 *26
	BVC label	If VF=0, PC+3+d8(label)→PC If VF=1, PC+3→PC	—	—	—	—	—	—	—	—	—	3	3/2	2/1	F5:FC:d8 *27
	BVS label	If VF=1, PC+3+d8(label)→PC If VF=0, PC+3→PC	—	—	—	—	—	—	—	—	—	3	3/2	2/1	F5:FD:d8 *28
	BNC label	If NF=0, PC+3+d8(label)→PC If NF=1, PC+3→PC	—	—	—	—	—	—	—	—	—	3	3/2	2/1	F5:FE:d8 *29
	BNS label	If NF=1, PC+3+d8(label)→PC If NF=0, PC+3→PC	—	—	—	—	—	—	—	—	—	3	3/2	2/1	F5:FF:d8 *30
	BRA label	PC+2+d8(label)→PC	—	—	—	—	—	—	—	—	—	2	2		EA:d8
Bccx	BEQX label	If ZX=1, PC+3+d8(label)→PC If ZX=0, PC+3→PC	—	—	—	—	—	—	—	—	—	3	3/2	2/1	F5:E8:d8 *31
	BNEX label	If ZX=0, PC+3+d8(label)→PC If ZX=1, PC+3→PC	—	—	—	—	—	—	—	—	—	3	3/2	2/1	F5:E9:d8 *32

※ Quick decoder : ON
(This setting cannot be made in this series.)

Notes:
20* src≥dest (lower 16 bits, signed)
21* src≤dest (lower 16 bits, signed)
22* src<dest (lower 16 bits, signed)
23* src>dest (lower 16 bits, unsigned)
24* src≥dest (lower 16 bits, unsigned)
25* src≤dest (lower 16 bits, unsigned)
26* src<dest (lower 16 bits, unsigned)
27* VF=0
28* VF=1
29* NF=0
30* NF=1
31* src=dest (24 bits)
32* src≠dest (24 bits)

Instruction	Mnemonic	Operation	OP EX.	Flag VX	CX	NX	ZX	VF	CF	NF	ZF	Code Size	Cycle	※ Cycle	Machine Code
Bccx	BLTX label	If (VX^NX)=1, PC+3+d8(label)→PC If (VX^NX)=0, PC+3→PC	—	—	—	—	—	—	—	—	—	3	3/2	2/1	F5:E0:d8 *33
	BLEX label	If ((VX^NX) | ZX)=1, PC+3+d8(label)→PC If ((VX^NX) | ZX)=0, PC+3→PC	—	—	—	—	—	—	—	—	—	3	3/2	2/1	F5:E3:d8 *34
	BGEX label	If (VX^NX)=0, PC+3+d8(label)→PC If (VX^NX)=1, PC+3→PC	—	—	—	—	—	—	—	—	—	3	3/2	2/1	F5:E2:d8 *35
	BGTX label	If ((VX^NX) | ZX)=0, PC+3+d8(label)→PC If ((VX^NX) | ZX)=1, PC+3→PC	—	—	—	—	—	—	—	—	—	3	3/2	2/1	F5:E1:d8 *36
	BCSX label	If CX=1, PC+3+d8(label)→PC If CX=0, PC+3→PC	—	—	—	—	—	—	—	—	—	3	3/2	2/1	F5:E4:d8 *37
	BLSX label	If (CX | ZX)=1, PC+3+d8(label)→PC If (CX | ZX)=0, PC+3→PC	—	—	—	—	—	—	—	—	—	3	3/2	2/1	F5:E7:d8 *38
	BCCX label	If CX=0, PC+3+d8(label)→PC If CX=1, PC+3→PC	—	—	—	—	—	—	—	—	—	3	3/2	2/1	F5:E6:d8 *39
	BHIX label	If (CX | ZX)=0, PC+3+d8(label)→PC If (CX | ZX)=1 PC+3→PC	—	—	—	—	—	—	—	—	—	3	3/2	2/1	F5:E5:d8 *40
	BVCX label	If VX=0, PC+3+d8(label)→PC If VX=1, PC+3→PC	—	—	—	—	—	—	—	—	—	3	3/2	2/1	F5:EC:d8 *41
	BVSX label	If VX=1, PC+3+d8(label)→PC If VX=0, PC+3→PC	—	—	—	—	—	—	—	—	—	3	3/2	2/1	F5:ED:d8 *42
	BNCX label	If NX=0, PC+3+d8(label)→PC If NX=1, PC+3→PC	—	—	—	—	—	—	—	—	—	3	3/2	2/1	F5:EE:d8 *43
	BNSX label	If NX=1, PC+3+d8(label)→PC If NX=0, PC+3→PC	—	—	—	—	—	—	—	—	—	3	3/2	2/1	F5:EF:d8 *44
JMP	JMP label16	PC+3+d16(label16)→PC	—	—	—	—	—	—	—	—	—	3	2	1	FC:d16-l:d16-h
	JMP label24	PC+5+d24(label24)→PC	—	—	—	—	—	—	—	—	—	5	4	3	F4:E0:d24-l:d24-m:d24-h
	JMP (An)	An→PC	—	—	—	—	—	—	—	—	—	2	3	2	F0:An<<2

※ Quick decoder : ON
(This setting cannot be made in this series.)

Notes:
33* src>dest (24 bits, signed)
34* src≥dest (24 bits, signed)
35* src≤dest (24 bits, signed)
36* src<dest (24 bits, signed)
37* src>dest (24 bits, unsigned)
38* src≥dest (24 bits, unsigned)
39* src≤dest (24 bits, unsigned)
40* src<dest (24 bits, unsigned)
41* VX=0
42* VX=1
43* NX=0
44* NX=1

Instruction	Mnemonic	Operation	OP EX.	Flag								Code Size	Cycle	❋ Cycle	Machine Code
				VX	CX	NX	ZX	VF	CF	NF	ZF				
JSR	JSR label16	A3-4→A3 PC+3→mem24(A3) PC+3+d16(label16)→PC	—	—	—	—	—	—	—	—	—	3	4	3	FD:d16-l:d16-h
	JSR label24	A3-4→A3 PC+5→mem24(A3) PC+5+d24(label24)→PC	—	—	—	—	—	—	—	—	—	5	5	4	F4:E1:d24-l:d24-m:d24-h
	JSR (An)	A3-4→A3 PC+2→mem24(A3) An→PC	—	—	—	—	—	—	—	—	—	2	5	4	F0:01+An<<2
NOP	NOP	PC+1→PC	—	—	—	—	—	—	—	—	—	1	1		F6
RTS	RTS	mem24(A3)→PC A3+4→A3	—	—	—	—	—	—	—	—	—	1	5		FE
RTI	RTI	mem16(A3)→PSW mem24(A3+2)→PC A3+6→A3	—	●	●	●	●	●	●	●	●	1	6		EB
PXST	PXST	Prefix instruction reversing the following instruction of addition/subtraction on saturation operation flag of PSW	—	—	—	—	—	—	—	—	—	2	2	1	F3:FC

❋ Quick decoder : ON
(This setting cannot be made in this series.)
Ver.2.01 (2000.5.16)

How to Read INSTRUCTION SET

■ Explanation of symbols used in the chart

Dn, Dm, Di	Data register
An, Am	Address register
MDR, PSW, PC	Multiplication and division register, program status word, program counter
imm8, imm16, imm16-l, imm16-h imm24, imm24-l, imm24-m, imm24-h	Constant
d8, d16, d16-l, d16-h d24, d24-l, d24-m, d24-h	Displacement
abs16, abs16-l, abs16-h abs24, abs24-l, abs24-m, abs24-h	Absolute address
mem8 (An), mem8 (abs16), mem8 (abs24)	8-bit memory data referenced at the address enclosed in parenthesis
mem16 (An), mem16 (abs16), mem16 (abs24)	16-bit memory data referenced at the address enclosed in parenthesis
mem24 (Am), mem24 (abs16), mem24 (abs24)	24-bit memory data referenced at the address enclosed in parenthesis
.bp,.lsb,.msb	Bit specification
&,I,^	Logical AND, logical OR, exclusive OR
˜,<<	Bit reversal, bit shift
VX, CX, NX, ZX	Extended overflow flag, extended carry flag, extended negative flag, extended zero flag
VF, CF, NF, ZF	Overflow flag, carry flag, negative flag, zero flag
temp	Temporary register inside CPU
→, ...	Assignment, reflection of computation results
H.M.	High-speed Multiplication

■ OP EX. (Operand Extension)

O	zero extension
S	sign extension
–	not applicable

■ Flag

●	change
—	no change
0	normally 0
1	normally 1
?	undefined

■ Code Size

Unit: byte

■ Cycle

The minimum number of cycles are specified.

Unit: machine cycle

a/b: there are branches in the 'a' cycle
there are no branches in the 'b' cycle

■ Machine Code

[:] separates the byte units. [<<2] indicates a 2-bit shift.

Dn, Dm, Di, An, Am: register numbers

D0	00	A0	00
D1	01	A1	01
D2	10	A2	10
D3	11	A3	11

■ Notes

- Instructions that access 16-bit and 24-bit data must use an even memory address.
- All 8-bit displacements (d8) and 16-bit displacements (d16) are sign extended.

MN102H00 SERIES INSTRUCTION MAP

<table>
<tr><th>First byte
Upper/Lower</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th></tr>
<tr><td>0</td><td colspan="16">MOV Dm, (An)</td></tr>
<tr><td>1</td><td colspan="16">MOVB Dm, (An)</td></tr>
<tr><td>2</td><td colspan="16">MOV (An), Dm</td></tr>
<tr><td>3</td><td colspan="16">MOVBU (An), Dm</td></tr>
<tr><td>4</td><td colspan="16">MOV Dm, (d8, An)</td></tr>
<tr><td>5</td><td colspan="16">MOV Am, (d8, An)</td></tr>
<tr><td>6</td><td colspan="16">MOV (d8, An), Dm</td></tr>
<tr><td>7</td><td colspan="16">MOV (d8, An), Am</td></tr>
<tr><td>8</td><td colspan="16">MOV Dn, Dm, (when src=dest,MOV imm8, Dn)</td></tr>
<tr><td>9</td><td colspan="16">ADD Dn, Dm</td></tr>
<tr><td>A</td><td colspan="16">SUB Dn, Dm</td></tr>
<tr><td>B</td><td colspan="4">EXTX Dn</td><td colspan="4">EXTXU Dn</td><td colspan="4">EXTXB Dn</td><td colspan="4">EXTXBU Dn</td></tr>
<tr><td>C</td><td colspan="4">MOV Dn, (abs16)</td><td colspan="4">MOVB Dn, (abs16)</td><td colspan="4">MOV (abs16),Dn</td><td colspan="4">MOVBU (abs16),Dn</td></tr>
<tr><td>D</td><td colspan="4">ADD imm8, An</td><td colspan="4">ADD imm8, Dn</td><td colspan="4">CMP imm8, Dn</td><td colspan="4">MOV imm16, An</td></tr>
<tr><td>E</td><td>BLT label</td><td>BGT label</td><td>BGE label</td><td>BLE label</td><td>BCS label</td><td>BHI label</td><td>BCC label</td><td>BLS label</td><td>BEQ label</td><td>BNE label</td><td>BRA label</td><td>RTI</td><td colspan="4">CMP imm16, An</td></tr>
<tr><td>F</td><td>Extended code A</td><td>Extended code B</td><td>Extended code C</td><td>Extended code D</td><td>Extended code E</td><td>Extended code F</td><td>NOP</td><td>Extended code G</td><td colspan="4">MOV imm16, Dn</td><td>JMP label16</td><td>JSR label16</td><td>RTS</td><td></td></tr>
</table>

Extended Code A

Second byte (Byte 1: F0)

<table>
<tr><th>Second byte
Upper/Lower</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th></tr>
<tr><td>0</td><td>JMP (A0)</td><td>JSR (A0)</td><td colspan="2"></td><td>JMP (A1)</td><td>JSR (A1)</td><td colspan="2"></td><td>JMP (A2)</td><td>JSR (A2)</td><td colspan="2"></td><td>JMP (A3)</td><td>JSR (A3)</td><td colspan="2"></td></tr>
<tr><td>1</td><td colspan="16"></td></tr>
<tr><td>2</td><td colspan="16">BSET Dm, (An)</td></tr>
<tr><td>3</td><td colspan="16">BCLR Dm, (An)</td></tr>
<tr><td>4</td><td colspan="16" rowspan="4">MOVB (Di, An), Dm</td></tr>
<tr><td>5</td></tr>
<tr><td>6</td></tr>
<tr><td>7</td></tr>
<tr><td>8</td><td colspan="16" rowspan="4">MOVBU (Di, An), Dm</td></tr>
<tr><td>9</td></tr>
<tr><td>A</td></tr>
<tr><td>B</td></tr>
<tr><td>C</td><td colspan="16" rowspan="4">MOVB Dm, (Di, An)</td></tr>
<tr><td>D</td></tr>
<tr><td>E</td></tr>
<tr><td>F</td></tr>
</table>

Extended Code B

Second byte (Byte 1: F1)

<table>
<tr><th>Second byte
Upper/Lower</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th></tr>
<tr><td>0</td><td colspan="16" rowspan="4">MOV (Di, An), Am</td></tr>
<tr><td>1</td></tr>
<tr><td>2</td></tr>
<tr><td>3</td></tr>
<tr><td>4</td><td colspan="16" rowspan="4">MOV (Di, An), Dm</td></tr>
<tr><td>5</td></tr>
<tr><td>6</td></tr>
<tr><td>7</td></tr>
<tr><td>8</td><td colspan="16" rowspan="4">MOV Am, (Di, An)</td></tr>
<tr><td>9</td></tr>
<tr><td>A</td></tr>
<tr><td>B</td></tr>
<tr><td>C</td><td colspan="16" rowspan="4">MOV Dm, (Di, An)</td></tr>
<tr><td>D</td></tr>
<tr><td>E</td></tr>
<tr><td>F</td></tr>
</table>

Extended Code C

Second byte (Byte 1: F2)

<table>
<tr><th>Second byte
Upper/Lower</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th></tr>
<tr><td>0</td><td colspan="16">ADD Dm, An</td></tr>
<tr><td>1</td><td colspan="16">SUB Dm, An</td></tr>
<tr><td>2</td><td colspan="16">CMP Dm, An</td></tr>
<tr><td>3</td><td colspan="16">MOV Dm, An</td></tr>
<tr><td>4</td><td colspan="16">ADD An, Am</td></tr>
<tr><td>5</td><td colspan="16">SUB An, Am</td></tr>
<tr><td>6</td><td colspan="16">CMP An, Am</td></tr>
<tr><td>7</td><td colspan="16">MOV An, Am</td></tr>
<tr><td>8</td><td colspan="16">ADDC Dn, Dm</td></tr>
<tr><td>9</td><td colspan="16">SUBC Dn, Dm</td></tr>
<tr><td>A</td><td colspan="16"></td></tr>
<tr><td>B</td><td colspan="16"></td></tr>
<tr><td>C</td><td colspan="16">ADD An, Dm</td></tr>
<tr><td>D</td><td colspan="16">SUB An, Dm</td></tr>
<tr><td>E</td><td colspan="16">CMP An, Dm</td></tr>
<tr><td>F</td><td colspan="16">MOV An, Dm</td></tr>
</table>

Extended Code D
Second byte(Byte 1: F3)

Second byte Upper/Lower	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	AND Dn, Dm															
1	OR Dn, Dm															
2	XOR Dn, Dm															
3	ROL Dn				ROR Dn				ASR Dn				LSR Dn			
4	MUL Dn, Dm															
5	MULU Dn, Dm															
6	DIVU Dn, Dm															
7																
8																
9	CMP Dn, Dm															
A																
B																
C	MOV D0, MDR	EXT D0			MOV D1, MDR	EXT D1			MOV D2, MDR	EXT D2			MOV D3, MDR	EXT D3		
D	MOV D0, PSW				MOV D1, PSW				MOV D2, PSW				MOV D3, PSW			
E	MOV MDR, Dn				NOT Dn											
F	MOV PSW, Dn												PXST		Extended code H	Extended code I

Extended Code E
Second byte (Byte 1: F4)

Second byte Upper/Lower	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	MOV Dm, (d24, An)															
1	MOV Am, (d24, An)															
2	MOVB Dm, (d24, An)															
3	MOVX Dm, (d24, An)															
4	MOV Dn, (abs24)				MOVB Dn, (abs24)							BSET imm8,(abs24)				BCLR imm8,(abs24)
5	MOV An, (abs24)															
6	ADD imm24, Dn				ADD imm24, An				SUB imm24, Dn				SUB imm24, An			
7	MOV imm24, Dn				MOV imm24, An				CMP imm24, Dn				CMP imm24, An			
8	MOV (d24, An), Dm															
9	MOVBU (d24, An), Dm															
A	MOVB (d24, An), Dm															
B	MOVX (d24, An), Dm															
C	MOV (abs24), Dn				MOVB (abs24), Dn				MOVBU (abs24), Dn							
D	MOV (abs24), An															
E	JMP label24	JSR label24		BSET imm8,(abs16)				BCLR imm8,(abs16)	BSET imm8, (d8,An)				BCLR imm8, (d8,An)			
F	MOV (d24, An), Am															

Extended Code F

Second byte(Byte 1: F5)

<table>
<tr><th>Second byte
Upper/Lower</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th></tr>
<tr><td>0</td><td colspan="4">AND imm8, Dn</td><td colspan="4">BTST imm8, Dn</td><td colspan="4">OR imm8, Dn</td><td colspan="4">ADDNF imm8, An</td></tr>
<tr><td>1</td><td colspan="16">MOVB Dm, (d8, An)</td></tr>
<tr><td>2</td><td colspan="16">MOVB (d8, An), Dm</td></tr>
<tr><td>3</td><td colspan="16">MOVBU (d8, An), Dm</td></tr>
<tr><td>4</td><td colspan="16">Extended Code J</td></tr>
<tr><td>5</td><td colspan="16">MOVX Dm, (d8, An)</td></tr>
<tr><td>6</td><td colspan="16">Extended Code K</td></tr>
<tr><td>7</td><td colspan="16">MOVX (d8, An), Dm</td></tr>
<tr><td>8</td><td colspan="8">TBZ(d8, A0) bp,label</td><td colspan="8">TBZ(d8, A1) bp,label</td></tr>
<tr><td>9</td><td colspan="8">BSET(d8, A0) bp</td><td colspan="8">BSET(d8, A1) bp</td></tr>
<tr><td>A</td><td colspan="8">TBNZ(d8, A0) bp,label</td><td colspan="8">TBNZ(d8, A1) bp,label</td></tr>
<tr><td>B</td><td colspan="8">BCLR(d8, A0) bp</td><td colspan="8">BCLR(d8, A1) bp</td></tr>
<tr><td>C</td><td colspan="8">TBZ(abs16) bp,label</td><td colspan="8">TBNZ(abs16) bp,label</td></tr>
<tr><td>D</td><td colspan="8">BSET(abs16) bp</td><td colspan="8">BCLR(abs16) bp</td></tr>
<tr><td>E</td><td>BLTX label</td><td>BGTX label</td><td>BGEX label</td><td>BLEX label</td><td>BCSX label</td><td>BHIX label</td><td>BCCX label</td><td>BLSX label</td><td>BEQX label</td><td>BNEX label</td><td colspan="2"></td><td>BVCX label</td><td>BVSX label</td><td>BNCX label</td><td>BNSX label</td></tr>
<tr><td>F</td><td colspan="8">Extended Code L</td><td colspan="4"></td><td>BVC label</td><td>BVS label</td><td>BNC label</td><td>BNS label</td></tr>
</table>

Extended Code G

Second byte (Byte 1: F7)

<table>
<tr><th>Second byte
Upper/Lower</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>A</th><th>B</th><th>C</th><th>D</th><th>E</th><th>F</th></tr>
<tr><td>0</td><td colspan="4">AND imm16, Dn</td><td colspan="4">BTST imm16, Dn</td><td colspan="4">ADD imm16, An</td><td colspan="4">SUB imm16, An</td></tr>
<tr><td>1</td><td>AND imm16 PSW</td><td colspan="3"></td><td>OR imm16 PSW</td><td colspan="3"></td><td colspan="4">ADD imm16, Dn</td><td colspan="4">SUB imm16, Dn</td></tr>
<tr><td>2</td><td colspan="4">MOV An, (abs16)</td><td colspan="12"></td></tr>
<tr><td>3</td><td colspan="4">MOV (abs16), An</td><td colspan="12"></td></tr>
<tr><td>4</td><td colspan="4">OR imm16, Dn</td><td colspan="4"></td><td colspan="4">CMP imm16, Dn</td><td colspan="4">XOR imm16, Dn</td></tr>
<tr><td>5</td><td colspan="16">MOVBU (d16, An), Dm</td></tr>
<tr><td>6</td><td colspan="16">MOVX Dm ,(d16, An)</td></tr>
<tr><td>7</td><td colspan="16">MOVX (d16, An), Dm</td></tr>
<tr><td>8</td><td colspan="16">MOV Dm, (d16, An)</td></tr>
<tr><td>9</td><td colspan="16">MOVB Dm, (d16, An)</td></tr>
<tr><td>A</td><td colspan="16">MOV Am, (d16, An)</td></tr>
<tr><td>B</td><td colspan="16">MOV (d16, An), Am</td></tr>
<tr><td>C</td><td colspan="16">MOV (d16, An), Dm</td></tr>
<tr><td>D</td><td colspan="16">MOVB (d16, An), Dm</td></tr>
<tr><td>E</td><td colspan="16"></td></tr>
<tr><td>F</td><td colspan="16"></td></tr>
</table>

Extended Code H
Third byte (Byte 1: F3, Byte 2 : FE)

Third byte Upper/Lower	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0																
1																
2																
3																
4																
5																
6																
7																
8																
9																
A																
B																
C	TBZ (abs24)bp,label								TBNZ (abs24)bp,label							
D	BSET (abs24)bp								BCLR (abs24)bp							
E																
F																

Extended Code I
Third byte (Byte 1: F3, Byte 2 : FF)

Third byte Upper/Lower	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0																
1																
2																
3																
4																
5																
6																
7																
8	TBZ (d8,A2)bp,label								TBZ(d8,A3)bp,label							
9	BSET(d8,A2)bp								BSET (d8,A3)bp							
A	TBNZ (d8,A2)bp,label								TBNZ(d8,A3)bp,label							
B	BCLR(d8,A2)bp								BCLR (d8,A3)bp							
C																
D																
E																
F																

Extended Code J
Third byte (Byte 1: F5, Byte 2 : 4n)

Upper/Lower	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	MULQL Dn,Dm	MULQH Dn,Dm														
1																
2																
3																
4																
5																
6																
7																
8																
9																
A																
B																
C																
D																
E																
F																

Extended Code K
Third byte (Byte 1: F5, Byte 2 : 6n)

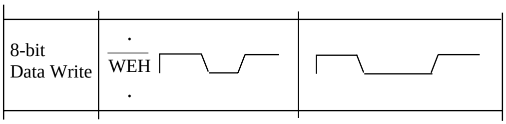
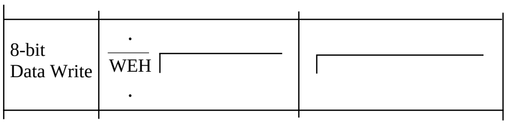
Upper/Lower	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0																
1	MULQ Dn,Dm															
2																
3																
4																
5																
6																
7																
8																
9																
A																
B																
C																
D																
E																
F																

Extended Code L
Third byte (Byte 1: F5, Byte 2 : Fn)

Upper/Lower	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0					MULQL imm8,Dn	MULQH imm8,Dn			MULQL imm16,Dn	MULQH imm16,Dn						
1																
2																
3																
4																
5																
6																
7																
8																
9																
A																
B																
C																
D																
E																
F																

Ver.1.01 (2000.5.16)

MN102H55D/55G/F55G User's Manual Record of Changes (Ver. 1.1 to Ver. 2.0)

Page	Line	Defini-tion	Former Version	New Version
13 to 21	Note	A		Unused pins require handling in the circuit (input pins are connected to VDD/VSS, output pins leave open, input/output pins are connected to VDD/VSS or leave open depending on pin direction setting).
22	Text Line 6	A		The unused pins require handling on the board. The input pins are connected to VDD or VSS. The output pins leave open. The lack of this handling causes the increase of current and unstable operation.
25	Table Line 4	C	Pulling the pin low sets 8-bit bus width while pulling the pin high sets 16-bit bus width.	Pulling the pin high sets 8-bit bus width while pulling the pin low sets 16-bit bus width.
39	Table	C	DUMX	DMUX
47	Table	C	... In addition, this pin can reads the pin state as the general-purpose input port PA6.	... In addition, this pin can reads the pin state as the general-purpose input port P76.
62	Section	A		The "2-1-3 Memory Connection Examples" is added.
65	Section	C	■ Example of DRAM (2CAS Method) Connection (16-bit Bus Width, 2 Wait)	■ Example of Burst ROM Connection (8-bit Bus Width, 4-3-3-3 Waits, Lower 2 bits of Address)
67	Section	A		The "2-1-4 Access to External Memory" is added.
68, 70, 72	Table	C	Table 2-1-4, Table 2-1-6, Table 2-1-8 	[Note:Pages 68, 70 and 72 are changed to pages 69,71,and 73.] Table 2-1-4, Table 2-1-6, Table 2-1-8 
69, 70	Table	C	A21-0	[Note:Pages 69 and 70 are changed to pages 70 and 71.] A23-0
71, 72	Table	C	A21-0	[Note:Pages 71 and 72 are changed to pages 72 and 73.] A22-8
73	Table	C	Table 2-1-9 Address/Data Separate Mode (16-bit Bus DRAM /UCAS and /LCAS Method)	[Note: Page 73 is changed to page 74.] Table 2-1-9 Address/Data Separate Mode (16-bit Bus Burst ROM Access)
74	Table	C	Table 2-1-10 Address/Data Separate Mode (8-bit Bus DRAM /UCAS and /LCAS Method)	Table 2-1-10 Address/Data Separate Mode (8-bit Bus Burst ROM Access)

<Definition> A: Add D: Delete C: Change

MN102H55D/55G/F55G User's Manual Record of Changes (Ver. 1.1 to Ver. 2.0)

<table>
<tr><th>Page</th><th>Line</th><th>Defini-tion</th><th>Former Version</th><th>New Version</th></tr>
<tr><td>75</td><td>Text Line 4</td><td>A</td><td></td><td>The $\overline{\text{RE}}$ short mode and the $\overline{\text{RE}}$ late mode do not affect the $\overline{\text{BSTRE}}$ pin connecting burst ROM.</td></tr>
<tr><td>82</td><td>Section</td><td>A</td><td></td><td>The "2-4 Activation Sequence" is added.</td></tr>
<tr><td>95,
98,
101</td><td>Note</td><td>C</td><td>Normally, the program generates the interrupt start address and then branches to that address.</td><td>After the program branches to x'080008', the program generates the interrupt service routine start address and then branches to that address.</td></tr>
<tr><td>100</td><td>Note</td><td>A</td><td></td><td>The watchdog interrupt is used to detect the CPU errors. The CPU cannot return to the previous operation before the watcchdog interrupt occurred after interrupt service routine is executed. Therefore, the CPU must reset after the watchdog interrupt occurred.</td></tr>
<tr><td>101</td><td>Text</td><td>A</td><td></td><td>■ Watchdog Timer in STOP Mode
When the watchdog timer is enabled and the CPU switches to STOP mode, the watchdog timer starts counting after it operates as the oscillation stabilization wait counter if the CPU returns to the previous mode (either NORMAL mode or SLOW mode) from STOP mode by an interrupt. In the MN102HF55G (Flash EEPROM version), 2^{17} must be selected as the watchdog interrupt cycle (WDM0='0', WDM1='1') when the CPU moves to STOP mode.</td></tr>
<tr><td>126</td><td>Text</td><td>C</td><td>These counters can serve as interval timers, event counters (in clock oscillation mode), one-phase PWMs, two-phase PWMs, one-phase captures, two-phase encoders (1x and 4x), ...</td><td>These counters can serve as interval timers, event counters (in clock oscillation mode), one-phase PWMs, two-phase PWMs, <u>two input captures</u>, two-phase encoders (1x and 4x), ...</td></tr>
<tr><td>133</td><td>Figure</td><td>C</td><td><table>
<tr><td>TMnMD [1:0]</td><td>TMnCA, TMnCB Operating Mode Selection</td></tr>
<tr><td>00</td><td>Compare register (single buffer)</td></tr>
<tr><td>01</td><td>Compare register (double buffer)</td></tr>
<tr><td>10</td><td>Capture register (TMnIOA pin high and low)</td></tr>
<tr><td>11</td><td>Capture register
(TMnIOA pin high, TMnIOB pin high)</td></tr>
</table></td><td><table>
<tr><td>TMnMD [1:0]</td><td>TMnCA, TMnCB Operating Mode Selection</td></tr>
<tr><td>00</td><td>Compare register (single buffer)</td></tr>
<tr><td>01</td><td>Compare register (double buffer)</td></tr>
<tr><td>10</td><td>Capture register (TMnIOA pin high <u>or</u> low)</td></tr>
<tr><td>11</td><td>Capture register
(TMnIOA pin <u>or</u> TMnIOB pin high)</td></tr>
</table></td></tr>
<tr><td>142</td><td>Text</td><td>C</td><td>3) When TM8BC=x'FFFF': all output waveforms consist of 1.</td><td>3) When <u>TM8CB</u>=x'FFFF': all output waveforms consist of 1.</td></tr>
<tr><td>156
157</td><td>Text</td><td>C</td><td>Figure 4-4-10 shows 000A-0007=0003, or 3 cycles.</td><td>Figure <u>4-4-12</u> shows 000A-0007=0003, or 3 cycles.</td></tr>
<tr><td>194</td><td>Note</td><td>A</td><td></td><td>In the duplex (half-duplex) asynchronous mode, both SBT pins become input when they are not selected to transmit, so they required pullup resistors.</td></tr>
</table>

<Definition> A: Add D: Delete C: Change

MN102H55D/55G/F55G User's Manual Record of Changes (Ver. 1.1 to Ver. 2.0)

<table>
<tr><th>Page</th><th>Line</th><th>Defini-
tion</th><th>Former Version</th><th>New Version</th></tr>
<tr><td>195</td><td>Note</td><td>A</td><td></td><td>Transmission/reception is possible within +/-2 % of baud rate errors.</td></tr>
<tr><td>208</td><td>Text</td><td>A
C</td><td>

■ **Data Transmission 2**
(6) Repeat steps (4) to (7) if the data is transmitted continuously.
■ **Stop Sequence**
(7) Write 0 to the SC3IIC flag of the SC3CTR register to end the data transmission. Do not write during transmission.
(8) Set the SCL3 pin output to high as soon as the SC3IIC flag is written. One cycle later, set the SDA3 pin output to high to start the stop sequence transmission. The SC3ISP flag of the SC3STR register becomes 1. (Reception must be enabled to detect the stop sequence.) Clear the SC3IST and SC3ISP flags of the SC3STR register by writing to or reading from the SC3TRB register.

</td><td>

(6) Read the dummy data of the serial 3 transmit/receive buffer (SC3TRB) after transmission ends.
(7) Verify that a parity error occurs by reading the serial 3 status register (SC3STR). When a parity error occurs, this means the slave responds normally. When a parity error does not occur, this means the slave does not respond. (This step is unnecessary for the system without ACK.)
■ **Data Transmission 2**
(8) Repeat steps (4) to (7) if the data is transmitted continuously.
■ **Stop Sequence**
(9) Write 0 to the SC3IIC flag of the SC3CTR register to end the data transmission. Do not write during transmission.
(10) Set the SCL3 pin output to high as soon as the SC3IIC flag is written. One cycle later, set the SDA3 pin output to high to start the stop sequence transmission. The SC3ISP flag of the SC3STR register becomes 1. (Reception must be enabled to detect the stop sequence.) Clear the SC3IST and SC3ISP flags of the SC3STR register by writing to or reading from the SC3TRB register.
(11) Set the SC3REN flag to disable once immediately after the stop sequence occurs.

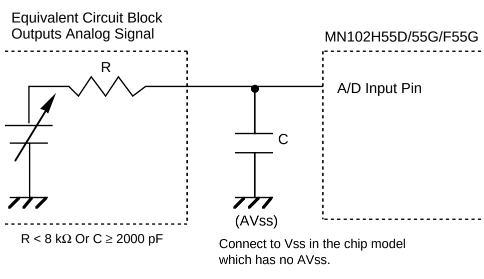
</td></tr>
<tr><td>209</td><td>Text

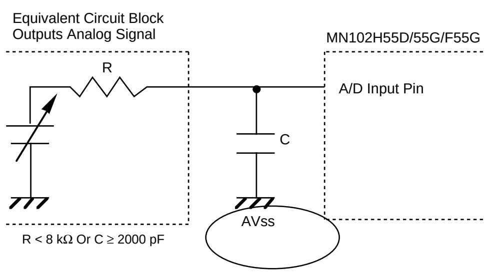
Note</td><td>C

A</td><td>
<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr>
<tr><td>SC3
TEN</td><td>SC3
REN</td><td>SC3
BRE</td><td>reser
ved</td><td>SC3
PTL</td><td>-</td><td>SC3
OD</td><td>reser
ved</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td></td><td>1</td><td>1</td></tr>
</table>
</td><td>
<table>
<tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr>
<tr><td>SC3
TEN</td><td>SC3
REN</td><td>SC3
BRE</td><td>reser
ved</td><td>SC3
PTL</td><td>-</td><td>SC3
OD</td><td>reser
ved</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td></td><td>1</td><td>1</td></tr>
</table>

Write the dummy data x'FF' always.

Verify that reception ends by an interrupt (either a serial 3 transmission end interrupt or a serial 3 reception end interrupt) or polling the received data flag of the serial 3 status register. Polling the reception busy flag is not allowed during I²C mode.

</td></tr>
<tr><td>212</td><td>Figure</td><td>D</td><td>


</td><td>


</td></tr>
<tr><td>225</td><td>Text</td><td>C</td><td>(4) Load the value of .....
TM3BR: x'00FE13'</td><td>(4) Load the value of .....
TM3MD: x'00FE23'</td></tr>
</table>

<Definition> A: Add D: Delete C: Change

MN102H55D/55G/F55G User's Manual Record of Changes (Ver. 1.1 to Ver. 2.0)

<table>
<tr><th>Page</th><th>Line</th><th>Defini-
tion</th><th>Former Version</th><th>New Version</th></tr>
<tr><td>234</td><td>Figure</td><td>C

A</td>
<td><table><tr><td>ATnBW</td><td>Transfer Unit</td></tr><tr><td>0</td><td>Byte</td></tr><tr><td>1</td><td>Word (disable when 8-bit bus width for external memory is selected)</td></tr></table></td>
<td><table><tr><td>ATnBW</td><td>Transfer Unit</td></tr><tr><td>0</td><td>Word (disable when 8-bit bus width for external memory is selected)</td></tr><tr><td>1</td><td>Byte</td></tr></table>
Do not activate ATC by an interrupt and write 0 to ATnEN flag by the user program simultaneously. Omitting this procedure causes the CPU to stop. Set the applicable register not to generate an interrupt for ATC activation factor before writing 0 to ATnEN flag.</td></tr>
<tr><td>239</td><td>Text</td><td>C</td>
<td><table><tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr><tr><td>AT0
EN</td><td>AT0
MD1</td><td>AT0
MD0</td><td>AT0
BW</td><td>AT0
DB8</td><td>AT0
DI</td><td>AT0
SB8</td><td>AT0
SI</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td></tr></table></td>
<td><table><tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr><tr><td>AT0
EN</td><td>AT0
MD1</td><td>AT0
MD0</td><td>AT0
BW</td><td>AT0
DB8</td><td>AT0
DI</td><td>AT0
SB8</td><td>AT0
SI</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td></tr></table></td></tr>
<tr><td>240</td><td>Text</td><td>C</td>
<td><table><tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr><tr><td>AT0
EN</td><td>AT0
MD1</td><td>AT0
MD0</td><td>AT0
BW</td><td>AT0
DB8</td><td>AT0
DI</td><td>AT0
SB8</td><td>AT0
SI</td></tr><tr><td>0
Busy flag indication</td><td>0
Hold</td><td>0
Hold</td><td>0
Hold</td><td>0
Hold</td><td>1
Hold</td><td>0
Hold</td><td>0
Hold</td></tr></table></td>
<td><table><tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr><tr><td>AT0
EN</td><td>AT0
MD1</td><td>AT0
MD0</td><td>AT0
BW</td><td>AT0
DB8</td><td>AT0
DI</td><td>AT0
SB8</td><td>AT0
SI</td></tr><tr><td>0
Busy flag indication</td><td>0
Hold</td><td>0
Hold</td><td>1
Hold</td><td>0
Hold</td><td>1
Hold</td><td>0
Hold</td><td>0
Hold</td></tr></table></td></tr>
<tr><td>244</td><td>Figure</td><td>C</td>
<td><table><tr><td>ETnBW</td><td>Transfer Unit</td></tr><tr><td>0</td><td>Byte</td></tr><tr><td>1</td><td>Word</td></tr></table></td>
<td><table><tr><td>ETnBW</td><td>Transfer Unit</td></tr><tr><td>0</td><td>Word</td></tr><tr><td>1</td><td>Byte</td></tr></table></td></tr>
<tr><td>248</td><td>Text</td><td>C</td>
<td><table><tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr><tr><td>ET0
FLG</td><td>-</td><td>ET0
MD0</td><td>ET0
BW</td><td>ET0
DB8</td><td>ET0
DI</td><td>ET0
SB8</td><td>ET0
SI</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td></tr></table></td>
<td><table><tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr><tr><td>ET0
FLG</td><td>-</td><td>ET0
MD0</td><td>ET0
BW</td><td>ET0
DB8</td><td>ET0
DI</td><td>ET0
SB8</td><td>ET0
SI</td></tr><tr><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td></tr></table></td></tr>
<tr><td>252</td><td>Text</td><td>C</td>
<td><table><tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr><tr><td>ET0
FLG</td><td>-</td><td>ET0
MD0</td><td>ET0
BW</td><td>ET0
DB8</td><td>ET0
DI</td><td>ET0
SB8</td><td>ET0
SI</td></tr><tr><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td></tr></table></td>
<td><table><tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr><tr><td>ET0
FLG</td><td>-</td><td>ET0
MD0</td><td>ET0
BW</td><td>ET0
DB8</td><td>ET0
DI</td><td>ET0
SB8</td><td>ET0
SI</td></tr><tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td></tr></table></td></tr>
<tr><td>256</td><td>Text</td><td>C</td>
<td><table><tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr><tr><td>ET0
FLG</td><td>-</td><td>ET0
MD0</td><td>ET0
BW</td><td>ET0
DB8</td><td>ET0
DI</td><td>ET0
SB8</td><td>ET0
SI</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td></tr></table></td>
<td><table><tr><td>15</td><td>14</td><td>13</td><td>12</td><td>11</td><td>10</td><td>9</td><td>8</td></tr><tr><td>ET0
FLG</td><td>-</td><td>ET0
MD0</td><td>ET0
BW</td><td>ET0
DB8</td><td>ET0
DI</td><td>ET0
SB8</td><td>ET0
SI</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td></tr></table></td></tr>
<tr><td>271</td><td>Figure</td><td>A</td><td></td><td>Add $\overline{\text{NMI}}$</td></tr>
<tr><td>275</td><td>Figure

Note</td><td>A

A</td><td></td><td>Add XO in pin name section and add STOP Control block diagram

The XI pin can be used as the port B1 when this pin is not used as the low-speed oscillation pin.</td></tr>
</table>

<Definition> A: Add D: Delete C: Change

MN102H55D/55G/F55G User's Manual Record of Changes (Ver. 1.1 to Ver. 2.0)

Page	Line	Definition	Former Version	New Version
281	Note	A		In the MN102H55D/55G, the ADB0CK and ADB1CK flags do not operate correctly. Compare the return address to the ADB0 register or ADB1 register to check the address break interrupt generation. The return address is the ADB0 register value plus 1 or the ADB1 register value plus 1.
283	Note	A		In the MN102H55D/55G, the ADB0CK and ADB1CK flags do not operate correctly. Compare the return address to the ADB0 register or ADB1 register to check the address break interrupt generation. The return address is the ADB0 register value plus 1 or the ADB1 register value plus 1.
392	List	C	15:14 Set Triger Conditions for K17 Pin Interrupt 13:12 Set Triger Conditions for K16 Pin Interrupt 11:10 Set Triger Conditions for K15 Pin Interrupt 9:8 Set Triger Conditions for K14 Pin Interrupt 7:6 Set Triger Conditions for K13 Pin Interrupt 5:4 Set Triger Conditions for K12 Pin Interrupt 3:2 Set Triger Conditions for K11 Pin Interrupt 1:0 Set Triger Conditions for K10 Pin Interrupt	15:14 Set Triger Conditions for $\overline{\mathrm{KI7}}$ Pin Interrupt 13:12 Set Triger Conditions for $\overline{\mathrm{KI6}}$ Pin Interrupt 11:10 Set Triger Conditions for $\overline{\mathrm{KI5}}$ Pin Interrupt 9:8 Set Triger Conditions for $\overline{\mathrm{KI4}}$ Pin Interrupt 7:6 Set Triger Conditions for $\overline{\mathrm{KI3}}$ Pin Interrupt 5:4 Set Triger Conditions for $\overline{\mathrm{KI2}}$ Pin Interrupt 3:2 Set Triger Conditions for $\overline{\mathrm{KI1}}$ Pin Interrupt 1:0 Set Triger Conditions for $\overline{\mathrm{KI0}}$ Pin Interrupt
393	List	C	7 Set OR Pin for K17 Pin 6 Set OR Pin for K16 Pin 5 Set OR Pin for K15 Pin 4 Set OR Pin for K14 Pin 3 Set OR Pin for K13 Pin 2 Set OR Pin for K12 Pin 1 Set OR Pin for K11 Pin 0 Set OR Pin for K10 Pin	7 Set OR Pin for $\overline{\mathrm{KI7}}$ Pin 6 Set OR Pin for $\overline{\mathrm{KI6}}$ Pin 5 Set OR Pin for $\overline{\mathrm{KI5}}$ Pin 4 Set OR Pin for $\overline{\mathrm{KI4}}$ Pin 3 Set OR Pin for $\overline{\mathrm{KI3}}$ Pin 2 Set OR Pin for $\overline{\mathrm{KI2}}$ Pin 1 Set OR Pin for $\overline{\mathrm{KI1}}$ Pin 0 Set OR Pin for $\overline{\mathrm{KI0}}$ Pin
398 401 404 407 410 413	List	C	12 Transfer Units 0: Byte 1: Word	12 Transfer Units 0: Word 1: Byte
478	List	D	15 DRAM (PSRAM) Operation for External Memory Space 3 14 DRAM (PSRAM) Operation for External Memory Space 2 13 DRAM (PSRAM) Operation for External Memory Space 1 12 DRAM (PSRAM) Operation for External Memory Space 0	15 DRAM Operation for External Memory Space 3 14 DRAM Operation for External Memory Space 2 13 DRAM Operation for External Memory Space 1 12 DRAM Operation for External Memory Space 0

<Definition> A: Add D: Delete C: Change

MN102H55D/55G/F55G User's Manual Record of Changes (Ver. 1.1 to Ver. 2.0)

<table>
<tr><th>Page</th><th>Line</th><th>Defini-tion</th><th>Former Version</th><th>New Version</th></tr>
<tr><td>479</td><td>List</td><td>C</td><td>15 DRAM Access Method Selection 0: 2WE method
1: 2CAS method</td><td>15 DRAM Access Method Selection 0: 2WE method
1: Reserved</td></tr>
<tr><td></td><td>Note</td><td>A</td><td></td><td>Use only 2WE method in DRAM mode. Do not use 2CAS method.</td></tr>
<tr><td>480</td><td>Note</td><td>A</td><td></td><td>The $\overline{RE}$ short mode and the $\overline{RE}$ late mode do not affect the $\overline{BSTRE}$ pin connecting burst ROM.</td></tr>
<tr><td>493</td><td>List</td><td>C</td><td>
<table>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>—</td><td>—</td><td>P7 IN5</td><td>P7 IN4</td><td>P7 IN3</td><td>P7 IN2</td><td>P7 IN1</td><td>P7 IN0</td></tr>
<tr><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td></tr>
<tr><td>0</td><td>0</td><td>Port</td><td>Port</td><td>Port</td><td>Port</td><td>Port</td><td>Port</td></tr>
<tr><td>0</td><td>0</td><td>0/1</td><td>0/1</td><td>0/1</td><td>0/1</td><td>0/1</td><td>0/1</td></tr>
</table>
5:0 Port 7 Input</td><td>
<table>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>—</td><td>P7 IN6</td><td>P7 IN5</td><td>P7 IN4</td><td>P7 IN3</td><td>P7 IN2</td><td>P7 IN1</td><td>P7 IN0</td></tr>
<tr><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td></tr>
<tr><td>0</td><td>Port</td><td>Port</td><td>Port</td><td>Port</td><td>Port</td><td>Port</td><td>Port</td></tr>
<tr><td>0</td><td>0/1</td><td>0/1</td><td>0/1</td><td>0/1</td><td>0/1</td><td>0/1</td><td>0/1</td></tr>
</table>
6:0 Port 7 Input</td></tr>
<tr><td></td><td>Note</td><td>A</td><td></td><td>Reading P76 pin identifys the status input $\overline{NMI}$.</td></tr>
<tr><td>494</td><td>List</td><td>C</td><td>
<table>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>—</td><td>PB IN1</td><td>PB IN0</td></tr>
<tr><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>Port</td><td>Port</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0/1</td><td>0/1</td></tr>
</table></td><td>
<table>
<tr><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>—</td><td>—</td><td>—</td><td>reserved</td><td>reserved</td><td>—</td><td>PB IN1</td><td>PB IN0</td></tr>
<tr><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td><td>R</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>Port</td><td>Port</td><td>0</td><td>Port</td><td>Port</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0/1</td><td>0/1</td><td>0</td><td>0/1</td><td>0/1</td></tr>
</table></td></tr>
<tr><td>511</td><td>List</td><td>C</td><td>4:2 P82 Input/Output Signal Switch 000: Port
001: TM0IO port, SBT3input
010: TM0IO output
011: SBT3 output
100: SBT3 half-duplex output
101: SBT3 open drain output
110: SBT2 input
(cannot use P21 as SBI2 input)</td><td>4:2 P82 Input/Output Signal Switch 000: Port
001: TM0IO port, SBT3input
010: TM0IO output
011: SBT3 output
100: SBT3 half-duplex output
101: SCL3 open drain output
110: SBT2 input
(cannot use P21 as SBI2 input)</td></tr>
<tr><td>512</td><td>List</td><td>C</td><td>7:5 P85 Input/Output Signal Switch 000:Port
001:TM9IOA port, SBT4 input
010:TM9IOA output
011:SBT4 output
100:SBT4 half-duplex output
101:SBT4 open drain output
110:SBO2 output</td><td>7:5 P85 Input/Output Signal Switch 000:Port
001:TM9IOA port, SBT4 input
010:TM9IOA output
011:SBT4 output
100:SBT4 half-duplex output
101:SCL4 open drain output
110:SBO2 output</td></tr>
<tr><td>522</td><td>Text</td><td>C</td><td>After reset, the initialization program must be located in the external memory space 0 (x'000000' to x'3FFFFF').</td><td>The initialization program must be executed first after reset release. The initialization program must be allocated on x'080000' in single-chip mode, memory expansion mode or processor mode.</td></tr>
<tr><td>530</td><td>Text</td><td>C</td><td>The pullup resistor value is 10 kΩ ±10 %.</td><td>The pullup resistor value is 10 kΩ ±1 kΩ .</td></tr>
<tr><td></td><td></td><td>C</td><td>Descrption of low-active pins /xxx</td><td>Descrption of low-active pina $\overline{xxx}$</td></tr>
</table>

<Definition> A: Add D: Delete C: Change

MN102H55D/55G/F55G

LSI User's Manual

September, 2000 2nd Edition

Issued by Matsushita Electric Industrial Co., Ltd.
Matsushita Electronics Corporation

Semiconductor Company, Matsushita Electronics Corporation

Nagaokakyo, Kyoto, 617-8520 Japan
Tel: (075) 951-8151
http://www.mec.panasonic.co.jp

SALES OFFICES

■ U.S.A. SALES OFFICE

Panasonic Industrial Company [PIC]

- **New Jersey Office:**
  2 Panasonic Way, Secaucus, New Jersey 07094
  Tel: 201-392-6173
  Fax: 201-392-4652
- **Milpitas Office:**
  1600 McCandless Drive, Milpitas, California 95035
  Tel: 408-945-5630
  Fax: 408-946-9063
- **Chicago Office:**
  1707 N. Randall Road, Elgin, Illinois 60123-7847
  Tel: 847-468-5829
  Fax: 847-468-5725
- **Atlanta Office:**
  1225 Northbrook Parkway, Suite 1-151,
  Suwanee, Georgia 30174
  Tel: 770-338-6940
  Fax: 770-338-6849
- **San Diego Office:**
  9444 Balboa Avenue, Suite 185
  San Diego, California 92123
  Tel: 619-503-2940
  Fax: 619-715-5545

■ CANADA SALES OFFICE

Panasonic Canada Inc. [PCI]

5700 Ambler Drive Mississauga, Ontario, L4W 2T3
Tel: 905-624-5010
Fax: 905-624-9880

■ GERMANY SALES OFFICE

Panasonic Industrial Europe G.m.b.H. [PIEG]

- **Munich Office:**
  Hans-Pinsel-Strasse 2 85540 Haar
  Tel: 89-46159-156
  Fax: 89-46159-195

■ U.K. SALES OFFICE

Panasonic Industrial Europe Ltd. [PIEL]

- **Electric component Group:**
  Willoughby Road, Bracknell, Berkshire RG12 8FP
  Tel: 1344-85-3773
  Fax: 1344-85-3853

■ FRANCE SALES OFFICE

Panasonic Industrial Europe G.m.b.H. [PIEG]

- **Paris Office:**
  270, Avenue de President Wilson
  93218 La Plaine Saint-Denis Cedex
  Tel: 14946-4413
  Fax: 14946-0007

■ ITALY SALES OFFICE

Panasonic Industrial Europe G.m.b.H. [PIEG]

- **Milano Office:**
  Via Lucini N19, 20125 Milano
  Tel: 2678-8266
  Fax: 2668-8207

■ TAIWAN SALES OFFICE

Panasonic Industrial Sales Taiwan Co.,Ltd. [PIST]

- **Head Office:**
  6th Floor, Tai Ping & First Building No.550. Sec.4,
  Chung Hsiao E. Rd. Taipei 10516
  Tel: 2-2757-1900
  Fax: 2-2757-1906
- **Kaohsiung Office:**
  6th Floor, Hsien 1st Road Kaohsiung
  Tel: 7-223-5815
  Fax: 7-224-8362

■ HONG KONG SALES OFFICE

Panasonic Shun Hing Industrial Sales (Hong Kong) Co., Ltd. [PSI(HK)]

11/F, Great Eagle Centre, 23 Harbour Road,
Wanchai, Hong Kong.
Tel: 2529-7322
Fax: 2865-3697

■ SINGAPORE SALES OFFICE

Panasonic Semiconductor of South Asia [PSSA]

300 Beach Road # 16-01
The Concourse Singapore 199555
Tel: 390-3688
Fax: 390-3689

■ MALAYSIA SALES OFFICE

Panasonic Industrial Company (Malaysia) Sdn. Bhd.

- **Head Office: [PICM]**
  Tingkat 16B Menara PKNS PJ No.17,Jalan Yong
  Shook Lin 46050 Petaling Jaya Selangor Darul Ehsan
  Malaysia
  Tel: 03-7516606
  Fax: 03-7516666
- **Penang Office:**
  Suite 20-17,MWE PLAZA No.8,Lebuh Farquhar,10200
  Penang Malaysia
  Tel: 04-2625550
  Fax: 04-2619989
- **Johore Sales Office:**
  39-01 Jaran Sri Perkasa 2/1,Taman Tampoi
  Utama,Tampoi 81200 Johor Bahru,Johor Malaysia
  Tel: 07-241-3822
  Fax: 07-241-3996

■ CHINA SALES OFFICE

Panasonic SH Industrial Sales (Shenzhen) Co., Ltd. [PSI(SZ)]

7A-107, International Business & Exhibition Centre,
Futian Free Trade Zone, Shenzhen 518048
Tel: 755-359-8500
Fax: 755-359-8516

Panasonic Industrial (Shanghai) Co., Ltd. [PICS]

1F, Block A, Development Mansion, 51 Ri Jing Street,
Wai Gao Qiao Free Trade Zone, Shanghai 200137
Tel: 21-5866-6114
Fax: 21-5866-8000

■ THAILAND SALES OFFICE

Panasonic Industrial (Thailand) Ltd. [PICT]

252/133 Muang Thai-Phatra Complex Building,31st
Fl.Rachadaphisek Rd.,Huaykwang,Bangkok 10320
Tel: 02-6933407
Fax: 02-6933423

■ KOREA SALES OFFICE

Panasonic Industrial Korea Co., Ltd. [PIKL]

Hanil Group Bldg.11th Fl.,191 Hangangro 2ga,
Youngsans-ku,Seoul 140-702,Korea
Tel: 82-2-795-9600
Fax: 82-2-795-1542

■ PHILIPPINES SALES OFFICE

National Panasonic Sales Philippines [NPP]

102 Laguna Boulevard Laguna Technopark Sta.
Rosa. Laguna 4026 Philippines
Tel: 02-520-3150
Fax: 02-843-2778

080600
Printed in JAPAN